EXHIBIT 3

Redacted Version of Document Provisionally Filed Under Seal

THE HONORABLE RICARDO S. MARTINEZ

IN THE UNITED STATES DISTRICT COURT

FOR THE WESTERN DISTRICT OF WASHINGTON

AT SEATTLE

Case No. 2:21-cv-00799-RSM-TLF

UTHERVERSE GAMING LLC,

Plaintiff,

v.

EPIC GAMES, INC.,

Defendant.

EXPERT REPORT OF CRAIG ROSENBERG, PH.D.

TABLE OF CONTENTS

Contents

Similar to the Travis Scott concert, the Ariana Grande concert was seen as a groundbreaking moment for virtual events within the gaming industry with a total of 27,000,000 millions viewers each tuning in to watch the performance.[38] The event demonstrated the potential for virtual concerts and events to bring people together from all around the world.

10 SUMMARY OF OPINIONS

I have been asked to opine whether Epic Games infringe on the asserted claims of the '071 patent and the '605 patent. It is my opinion that claims 8 and 10 of the '071 patent and claims 2, 5, and 8 of the '605 patent are infringed on by Epic Games, Inc., by the design, creation, production, and presentation of the four accused events (Marshmello, Star Wars: The Rise of Skywalker, Travis Scott, and Ariana Grande) and the various Epic technologies that were used in the design, creation, production, and the presentation of the four accused events. These technologies include any custom software development associated with the design, creation, production, and presentation of the four accused events, as well as the Fortnite gaming environment and Unreal Engine used in the creation, production, and presentation of the four accused events.

11 KEY CONCEPTS AND TECHNOLOGY BACKGROUND

It should be noted I have reviewed both UE4 and UE5 documentation in my analysis for this report. As far as the extent of any such documentation is cited herein, I am not aware of any differences in the documentation for UE4 versus that of UE5 or the testimony of Mr. Imbriaco or Mr. Axt that affect my opinions as set forth in this report.

The following are the definitions of various terms that may be used within this report.

11.1 Terms

Actors: An Actor is any object that can be placed into a level, such as a Camera, static mesh, or player start location. Actors support 3D transformations such as translation, rotation, and scaling.

[38] "Top 10 Most Popular Metaverse Concerts," https://www.metaversemarcom.io/post/top-10-most-popular-metaverse-concerts

8) As described in this report Epic utilizes a common environment, called the Lobby. All players start in the Lobby which is a common environment. Players are moved from the Lobby to a specific instance (parallel dimension) using the matchmaking process. When Epic says, "Avatars assigned into an instance spawn directly into that instance; they do not access it from another area within the game environment," this is misleading as the players do access the instance (parallel dimension) from the Lobby which is *"another common area within the game environment."*

16 INFRINGEMENT OF THE '605 PATENT BY TRAVIS SCOTT

16.1 Claim 1[i]

"A method of playing back a recorded experience in a virtual worlds system, comprising:"

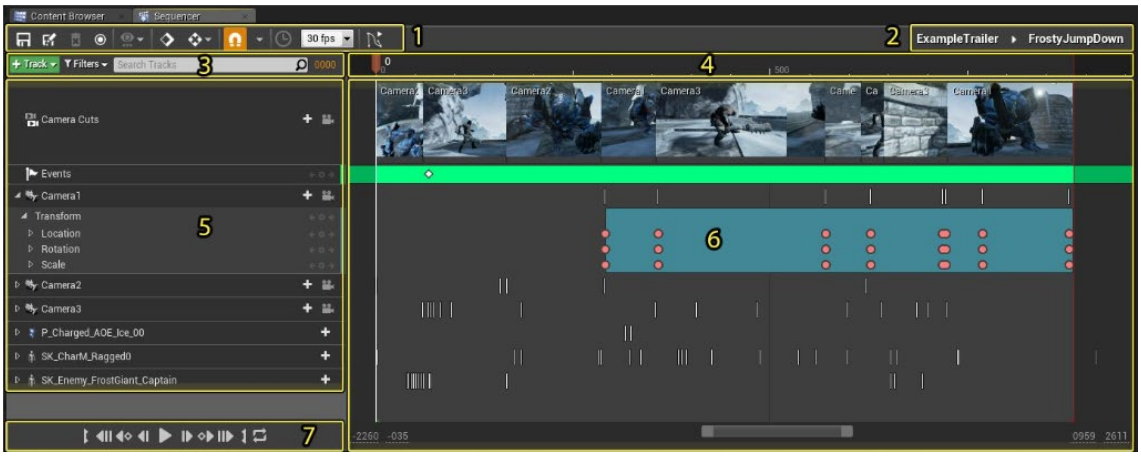
As an initial matter, in the Court's claim construction order, the Court said, "After considering the briefing of the parties and arguments at the hearing, the Court agrees with Utherverse and declines to rule that the preambles are limiting."[457]

Playing back a recorded experience implicitly requires that the experience is first recorded. As described above, Epic Games has created the Sequencer tool that allows for the creation of sequences that are stored in a recorded experience file. In Unreal Engine, sequences can be created and played back with the use of the Sequencer, which is a powerful cinematic editing tool that enables the creation of real-time, animations, in-engine cutscenes, cinematics, and scripted events.[458] Sequencer allows for the creation of complex sequences by combining

[457] Order regarding Claims Construction, page 15
[458] https://docs.unrealengine.com/4.26/en-US/AnimatingObjects/Sequencer/Overview/

various elements such as animations, audio, camera shots, and lighting in a linear or non-linear fashion.[459]

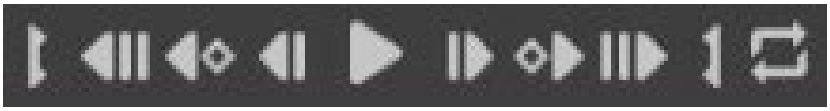
The image below, from the Unreal Engine documentation, shows the various parts of the Sequencer Editor including; 1) Tool Bar, 2) Sequence Breadcrumbs, 3) Add/Filter Tracks, 4) Timeline, 5) Tree View, 6) Tracks Area, and 7) Playback Controls.[460]



This section of the Epic Games documention on the Unreal Engine Sequencer describes, among other things, how the Sequencer Editor is used to both create animations and playback animations. The documentation shows using the CurveEditor to fine tune animation keys that are placed in the Tracks Area as well as adjusting properites of objects in the tracks area of the Sequencer such as start offset, end offset, play rate, etc.

In addition to the recording of animations, the Unreal Engine Sequencer supports playback of animations as well (see section 7 in the image above.) The Playback Controls section of the documentation discusses multiple features to assist animators in playing back the recorded experience.[461] These features include the following commands and controls.

[459] https://www.youtube.com/watch?v=D3trMpkxAFk
[460] https://docs.unrealengine.com/4.26/en-US/AnimatingObjects/Sequencer/ReferenceEditor/
[461] https://docs.unrealengine.com/4.26/en-US/AnimatingObjects/Sequencer/ReferenceEditor/



Command	Description
	Set the playback start position to the current position indicated by the time marker.
	Jump to the playback start position.
	Jump to the previous key in the selected track(s).
	Jump to the previous frame.
	Play or Pause the Level Sequence from the position of the time marker.
	Jump forward a frame.
	Jump to the next key in the selected track(s).
	Jump to the playback end position.
	Set the playback end position to the current position indicated by the time marker.
/	Toggles between looping the Level Sequence during playback.

In addition, the Sequencer supports the ability to manipulate the animation playback rate through the use of a Play Rate Track.[462] Through the use of the Unreal Engine Sequencer, Epic Games has the ability to both record animations at design time, and then playback the animations at either design time and/or at run-time.

[462] https://docs.unrealengine.com/4.26/en-US/AnimatingObjects/Sequencer/HowTo/TracksPlayRate/

Unreal Engine assets, including animation sequence data that represent the recorded experience file are stored in binary files that end with the extension ".uasset".[463] This is the native Unreal Engine file format for Unreal Engine assets. For example, the file [redacted] (bates EPIC-SRC101), is an asset file that holds the animation of the mannequin of Travis Scott.

This binary file has the data that determines the movements and motions of the Travis Scott avatar that was played back in the same way for each of the five Travis Scott virtual performances that took place between April 23rd, 2020 and April 25th, 2020.[464] [465] [466]It should be noted that the Epic Games technologies, which includes the Sequencer of the Unreal Engine, allowed for both the creation and playing back of the animation of the mannequin of Travis Scott.

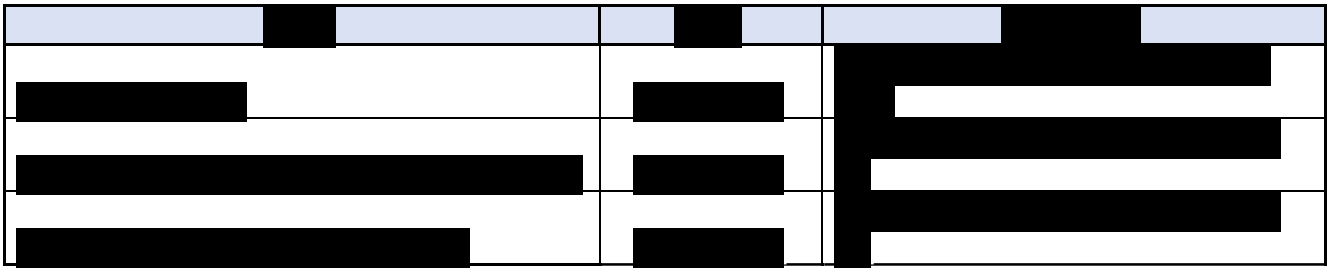
The animation of the mannequin of Travis Scott was a recorded experience using the Unreal Engine Sequencer and it was encapsulated in one or more files (e.g., [redacted] and it was played back in the same way for each of the five performances.[467] Mr. Mark Imbriaco described a triggering event that started the sequence that "*refers to the preprogrammed – think of it as the choreography of the event that is predesigned that exists on the server and the client already.*"[468] He goes on to say, "*When the live events start, they're not truly live. I mean, they are live, they're – but they're being driven by this preprogrammed sequence of movements that are designed using a technology called the sequencer.*"[469] The Epic Games technology of the Unreal Engine Sequencer, combined with the Fortnite environment provided, "A method of playing back a recorded experience in a virtual worlds system."[470]

---

[463] Deposition Testimony of Peter Axt, 221:12-13
[464] https://www.fortnite.com/news/astronomical
[465] EPIC00014958 – Cyclone-Jerky Run of Show
[466] Deposition Testimony of Mr. Peter Axt, 298:13-18
[467] Deposition Testimony of Mr. Peter Axt, 111:12-112:4, 294:24-298:18
[468] Deposition Testimony of Mr. Mark Imbriaco, 314:11-25
[469] Deposition Testimony of Mr. Mark Imbriaco, 315:1-5
[470] 605 Patent, Claim 1[i]

In addition to the recorded experience file of [redacted] bates EPIC-SRC101), below is a representative blueprint file that is responsible for playing back the recorded experience file.

[redacted]



16.1.1 Epic's Non-Infringement Contentions

Epic in their Third Amended Non-Infringement Chart for '605 (Exhibit G, page 1), states the following:

To the extent the preamble is limiting, Epic denies that the Travis Scott Concert involved a "method of playing back a recorded experience in a virtual worlds system." The Travis Scott Concert was not a "playing back" of a "recorded experience," but rather an interactive, preprogrammed sequence animated in real-time.

I disagree with Epic on their other assertions for the following reasons:

[471] Deposition Testimony of Peter Axt, 221:12-13
[472] Deposition Testimony of Peter Axt, 224:2-7

1) Playing back a recorded experience implicitly requires that the experience is first recorded. As described above, Epic Games has created the Sequencer tool that allows for the creation of sequences that are stored in a recorded experience file. In Unreal Engine, sequences can be created and played back with the use of the Sequencer, which is a powerful cinematic editing tool that enables the creation of real-time, animations, in-engine cutscenes, cinematics, and scripted events.[473] Sequencer allows for the creation of complex sequences by combining various elements such as animations, audio, camera shots, and lighting in a linear or non-linear fashion.[474]
2) This section of the Epic Games documention on the Unreal Engine Sequencer describes, among other things, how the Sequencer Editor is used to both create animations and playback animations.[475] The Playback Controls section of the documentation discusses multiple features to assist animators in playing back the recorded experience.[476]
3) As described in this report, the movements of various elements in the Travis Scott concerts were preprogrammed using the keyframe animation in the Sequencer of Unreal Engine. These recordings were played back in the same way for each of the five Travis Scott concerts that took place between April 23rd, 2020 and April 25th, 2020.[477] The Unreal engine Sequencer created a recorded experience file [redacted] (bates EPIC-SRC101) that was used to playback the animation of the mannequin of Travis Scott in the same way each time the concert was broadcast.

16.2 Claim 1[ii]

"instantiating, using one or more processors of a server, a new instance of a scene, the new instance being defined by data stored in memory, at least one client device displaying and participating in the new instance;"

---

[473] https://docs.unrealengine.com/4.26/en-US/AnimatingObjects/Sequencer/Overview/
[474] https://www.youtube.com/watch?v=D3trMpkxAFk
[475] https://docs.unrealengine.com/4.26/en-US/AnimatingObjects/Sequencer/ReferenceEditor/
[476] https://docs.unrealengine.com/4.26/en-US/AnimatingObjects/Sequencer/ReferenceEditor/
[477] Deposition Testimony of Mr. Peter Axt, 298:13-18

First, it should be noted that Epic admits infringement of this claim limitation. In their Third Amended Non-Infringement Contentions, Epic states, "*Epic admits that the Travis Scott Concert involved creating new instances of a scene, and that players participated in the instances using client devices*."[478]

Multiple Epic technologies were utilized to accomplish this, as discussed below.

"instantiating, using one or more processors of a server, a new instance of a scene, the new instance being defined by data stored in memory"

The plurality of parallel dimensions in a computer memory were provided by provisioning fleets of virtual machines using EC2 and GCE instances built from a single VM image.[479] A virtual machine has all of the same capabilities of a real computer including processing, memory, storage, networking, etc.[480] [481] The section above on Autoscaling, Dedicated Servers, Master Control Program (MCP), and Matchmaking Service (MMS) above explains how Epic Games utilizes multiple servers to ensure that the events run smoothly, with minimal latency in order to account for variable and fluctuating user demand.

Each new instance of a server virtual machine gets a copy of a master virtual image. This is described by Epic in describing how they use fleets of dedicated servers that are 100% identical to one another. Fleets are resized by adding and removing compute instances (EC2 or GCE instances built from a single VM Image) from the fleet.[482] In this way, each of the virtual machine instances are duplicates of one another and Epic's own internal documents say that they are "100% identical to one another."[483] Mr. Imbriaco also said that the dedicated server processes all come from the same installation package.[484] He also said that fleets are resized by

[478] Third Amended Non-Infringement Chart of '605 Patent – Exhibit G – Page 1
[479] EPIC-00020439 (page 2)
[480] Deposition Testimony of Mark Imbriaco, 212:8-12
[481] Deposition Testimony of Peter Axt, 23:19-24:23
[482] EPIC-00020439
[483] EPIC-00020439
[484] Deposition Transcript of Mark Imbriaco, 203:14-204:11

adding and removing instances built from the same virtual computer image.[485] He also said that it is the Fortnite developers that create the installation packages.[486] He also said that the VM image has the Fortnite version is stored in the build source that is created by Epic Games developers.[487]

On the server side, each instance is its own virtual machine with its own virtual memory that is used to host a small subset of participants. Each virtual machine has its own dedicated virtual memory and each virtual machine is located within one physical server that has its own physical memory.[488] [489] [490] On the client side, each client device also has its own memory as well. As discussed previously, the client device can be a PC, a Mac, or one of several gaming consoles including Sony PlayStation, Microsoft Xbox, Nintendo Switch, as well as Android devices and iOS devices.[491] [492]

"at least one client device displaying and participating in the new instance;"

The first step is that when the user is in the Lobby, they can utilize the user interface to [redacted] request to matchmake [redacted]

[485] Deposition Transcript of Mark Imbriaco, 205:3-207:25
[486] Deposition Transcript of Mark Imbriaco, 117:2-18
[487] Deposition Transcript of Mark Imbriaco, 262:1-20
[488] Deposition Testimony of Mark Imbriaco, 212:8-12
[489] Deposition Testimony of Peter Axt, page 23:19-24:23
[490] https://aws.amazon.com/what-is/virtualization/
[491] https://www.epicgames.com/help/en-US/fortnite-c5719335176219/technical-support-c5719372265755/what-platforms-or-devices-are-compatible-with-fortnite-a5720385202331
[492] https://www.fortnite.com/mobile
[493] Deposition Transcript of Peter Axt at 236:6-21
[494] *Id.*
[495] *Id.*
[496] Deposition Transcript of Peter Axt at 237:14-19
[497] Deposition Transcript of Peter Axt at 237:11-14

[redacted]

Given that there was a maximum number of [redacted] per dedicated server session for the Travis Scott and Ariana Grande concert, players could only be assigned to a dedicated server session if there were fewer than [redacted].[501] It is not possible or a dedicated server to receive more than its configured maximum number of players.[502] [redacted]

Once a given dedicated server is full (e.g., [redacted]), the matchmaking is complete for that dedicated server.[505]

[redacted]

[498] Deposition Transcript of Peter Axt at 239:2-12
[499] Deposition Transcript of Peter Axt at 241:17-242:11
[500] Deposition Transcript of Peter Axt at 243:9-244:21
[501] Deposition Transcript of Peter Axt at 245:23-246:22
[502] Deposition Transcript of Peter Axt at 247:5-21
[503] Deposition Transcript of Peter Axt at 249:17-250:3
[504] Deposition Transcript of Peter Axt at 250:4-250:19
[505] Deposition Transcript of Peter Axt at 251:11-252:22
[506] Deposition Transcript of Peter Axt at 252:3-22
[507] *Id.*
[508] Deposition Transcript of Peter Axt at 255:2-18

[redacted]

Mr. Mark Imbriaco explains that information about the world is stored using a replication graph. He describes the replication graph as being *"the piece of technology that is used to determine what changes in game state are relevant to each player at a point in time, and provide those updates to players, as needed, so that they have a reasonably accurate representation of the game state that's relevant to them"* and that the replication graph *"includes position of objects, characters, and other parts of game state."* [510] The game state information is delivered through the replication graph and this would include the four accused events.[511] Mr. Imbriaco say that it is the server that distributes game-state changes to the clients.[512] [513] The Fortnite environment that is running on the client device is responsible playing back and rendering by getting any needed the assets from server (via the replication graph), and rendering it on the client device.[514] As discussed previously, the client device can be a PC, a Mac, or one of several gaming consoles including Sony PlayStation, Microsoft Xbox, Nintendo Switch, as well as Android devices and iOS devices.[515] [516]

In the way described above, Epic Games creates and populates new instances so that the multiple client devices are able to connect to these new instances so they can participate in these new instances, get updates from these instances, and display the information that comes from each of the new instances.

---

[509] Deposition Transcript of Peter Axt at 256:5-24
[510] Deposition Testimony of Mark Imbriaco, 113:7-13, 113:17-18
[511] Deposition Testimony of Mark Imbriaco, 122:7-13
[512] Deposition Testimony of Mark Imbriaco, 126:11-15
[513] Deposition Testimony of Mark Imbriaco, 142:2-11
[514] Deposition Testimony of Mark Imbriaco, page 118
[515] https://www.epicgames.com/help/en-US/fortnite-c5719335176219/technical-support-c5719372265755/what-platforms-or-devices-are-compatible-with-fortnite-a5720385202331
[516] https://www.fortnite.com/mobile

Below are several representative source code files and blueprints that were used in assigning participants to new instances (e.g., [redacted] and reservations during the matchmaking process (e.g., [redacted]) for the Travis Scott concert.

[redacted]

[redacted]

[redacted]

[redacted]

[redacted]

[redacted]

[redacted]

[redacted]

16.2.1 Epic's Non-Infringement Contentions

Epic in their Third Amended Non-Infringement Chart for '605 (Exhibit G, page 1), states the following:

Epic admits that the Travis Scott Concert involved creating new instances of a scene, and that players participated in the instances using client devices.

I agree with Epic that, "*the Travis Scott Concert involved creating new instances of a scene, and that players participated in the instances using client devices.*"

As Epic already agrees that it infringes Claim 1[ii] of the '605 patent, I will keep this analysis short. Previously in this report, I describe how Epic uses technologies of Autoscaling, Dedicated Servers, Master Control Program (MCP), and Matchmaking Service (MMS) to create a fleet of servers and then connect client devices to a specific server. I also describe how the

[517] https://docs.unrealengine.com/4.26/en-US/API/Plugins/OnlineSubsystemUtils/APartyBeaconHost/

virtual machines that are created have virtual memory that are assigned to them and these virtual machines reside on physical servers that have physical memory.[518] [519] [520]

16.3 Claim 1[iii]

"retrieving a recorded experience file from the memory, the recorded experience file having been generated by saving an initial scene state and saving subsequent changes and respective times during a time period of the recorded experience;"

"retrieving a recorded experience file from the memory"

As described above, the Unreal Engine Sequencer has the ability to create recorded experiences and save them in a file (e.g., the recorded experience file.) In the case of the Travis Scott virtual concerts, the file "[redacted]" (bates EPIC-SRC101) is the Unreal Engine proprietary binary asset file that stores animation data for the mannequin of Travis Scott, so that the mannequin moves and gestures in the same way each time that the recorded experience file is played back.[521]

In order to utilize the [redacted] file, when loading the Unreal Engine project, Epic software would retrieve this file from non-volatile memory (e.g., SSD or spinning hard disk) and copy it into volatile memory (e.g., RAM.) Once the file was in volatile memory (e.g., RAM), the computer processor could retrieve this file from RAM for use. Both the retrieving of the recorded experience file from non-volatile memory as well as the retrieving the file from RAM would constitute "retrieving a recorded experience file from the memory."

"the recorded experience file having been generated by saving an initial scene state"

With regard to "the recorded experience file having been generated by saving an initial scene state," the Unreal Engine Sequencer allows for the saving of an initial scene state. A

---

[518] Deposition Testimony of Mark Imbriaco, 212:8-12
[519] Deposition Testimony of Peter Axt, 23:19-24:23
[520] https://aws.amazon.com/what-is/virtualization/
[521] Deposition Testimony of Mr. Peter Axt, 298:10-18

central component of the Unreal Engine Sequencer is the timeline. The timeline supports defining an initial scene state.[522]

The timeline supports both a green start marker and a red end marker. The start time is typically at time zero. This represents the initial state of the scene, or "the initial scene state." The user of the Unreal Engine Sequencer and timeline is free to set and save any initial attributes of the scene that they like at the beginning of the timeline. Examples of attributes that can be set for the initial scene might include the X, Y, and Z position of objects in the scene, the X, Y, and Z rotation of objects in the scene, the scaling of objects in the scene, the color of objects in the scene, various shader and lighting attributes of objects in the scene, data associated with a specific poses or positions of an avatar, etc.

"and saving subsequent changes and respective times during a time period of the recorded experience;"

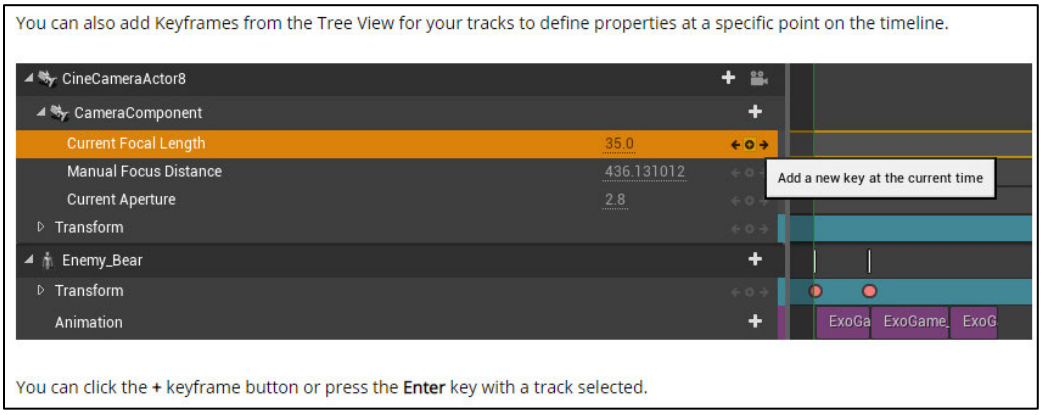
The Unreal Engine Sequencer supports keyframe animation. Keyframe animation in Unreal Engine refers to the process of creating animations by defining specific poses or transformations for 3D objects at certain points in time, known as keyframes. These keyframes are then interpolated by the engine to generate all of the in-between frames, thus creating a smooth animation sequence.

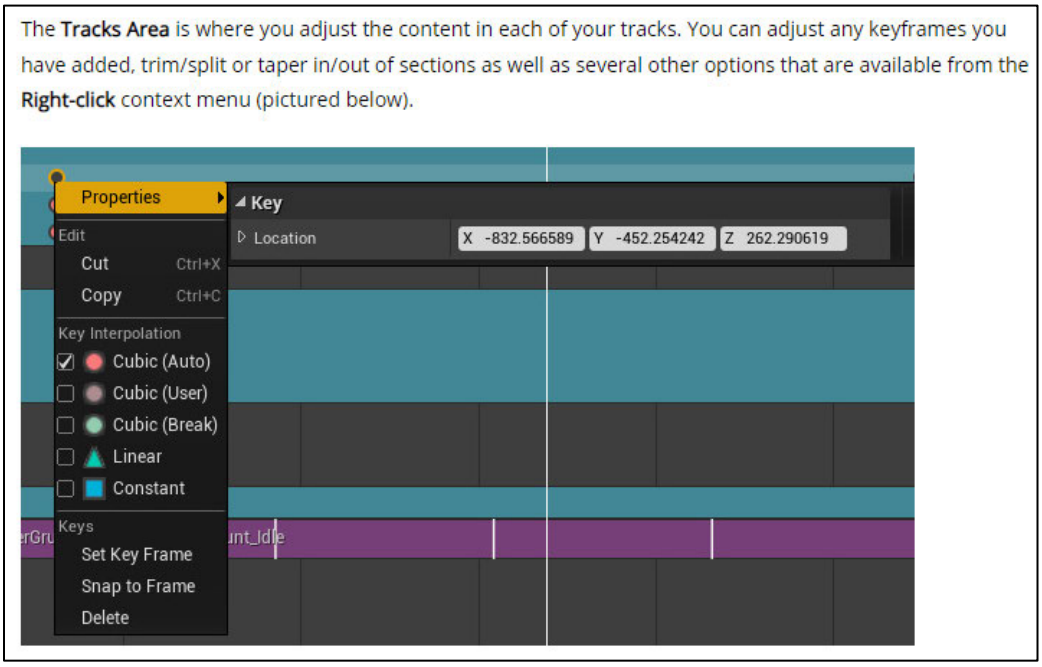
The Unreal Engine Sequencer allows users to create animations by setting keyframes for various properties like position, rotation, and scale of 3D objects, as well as cameras, lights, attributes of specific poses and positions of avatars, and other elements at specific points in time in the animation timeline. The properties that are set at the beginning of the timeline specify the initial scene state and properties can be set at later times using keyframes to set the subsequent changes to the animation at specific times. In addition, users can control attributes of the timing and the interpolation between keyframes in order to create smooth animations.

---

[522] https://docs.unrealengine.com/4.26/en-US/AnimatingObjects/Sequencer/ReferenceEditor/

The following is from the Unreal Engine documentation that describes how keyframes are added at subsequent points on the timeline.[523]

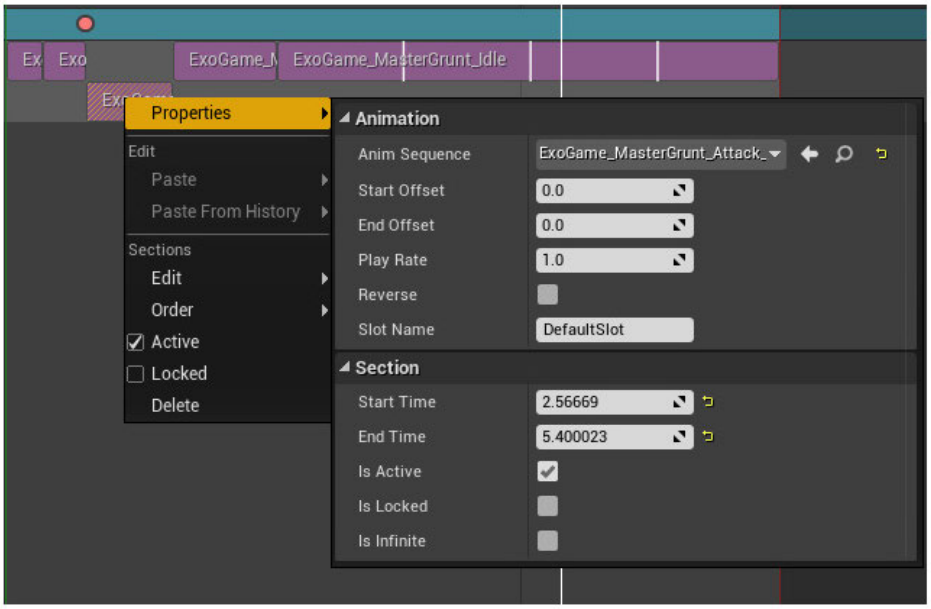




[523] https://docs.unrealengine.com/4.26/en-US/AnimatingObjects/Sequencer/ReferenceEditor/

Above, right-clicking a keyframe for a Transformation track gives the ability to change the key interpolation type as well as other options that can be adjusted from the **Properties** rollout, which updates based on the type of content your Right-click on.

Below, right-clicking on an animation assigned to an Animation Track and selecting properties gives us the ability to change the currently assigned animation as well as adjust properties related to the animation itself, such as start/end offsets, play rate, and more.

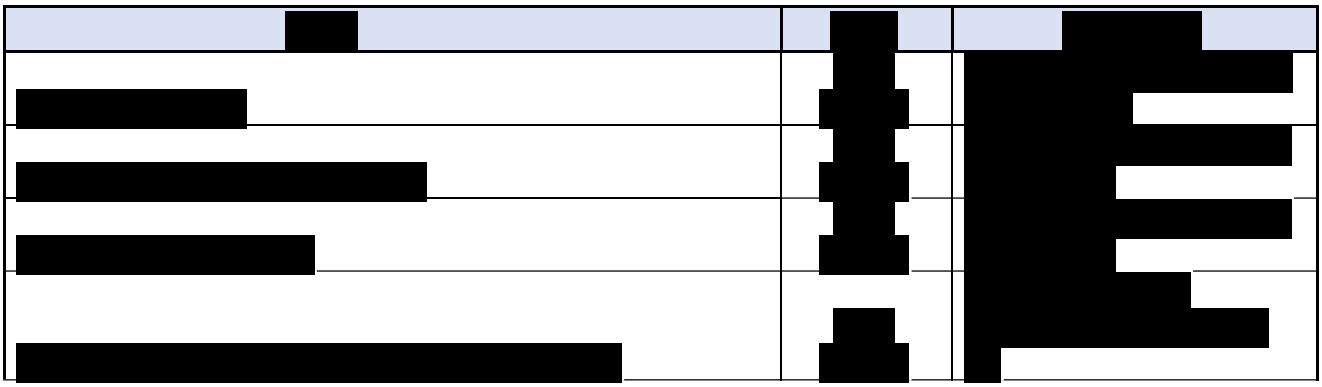


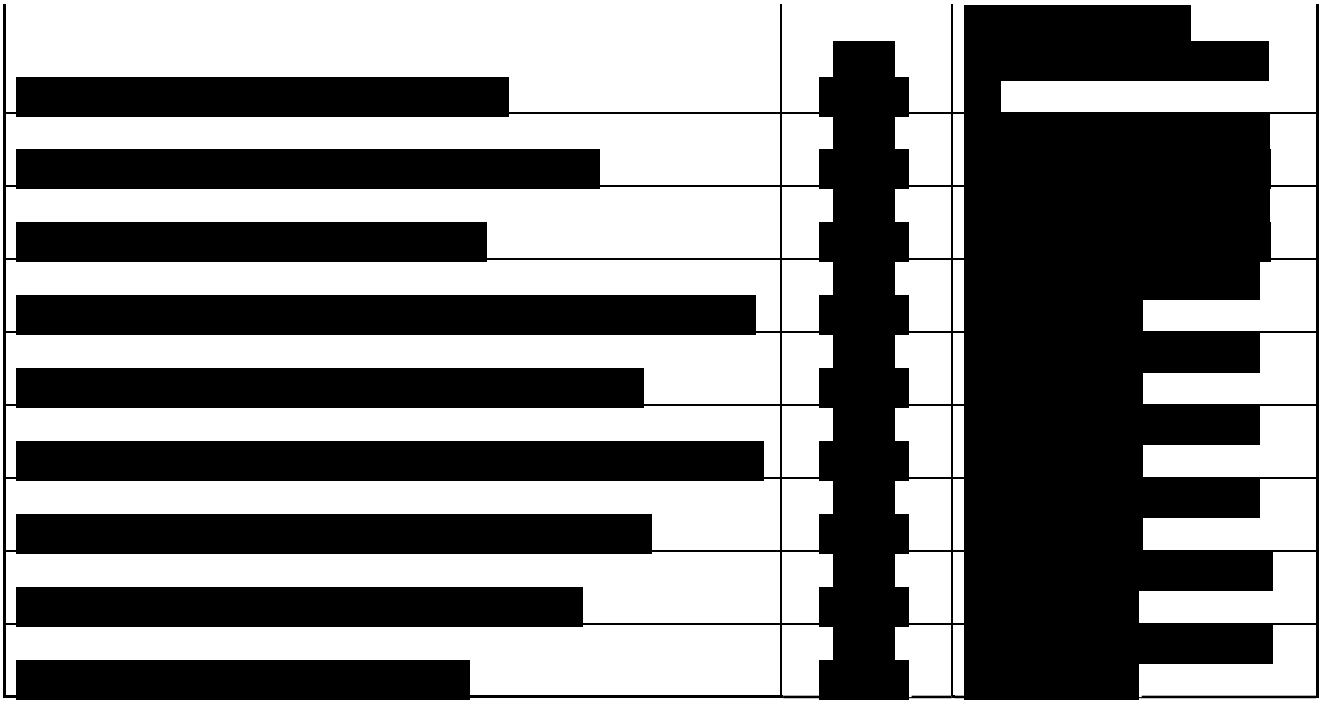
By setting keyframes on the Sequencer timeline, the Epic designers and implementers of the Travis Scott concerts were able to save both the initial scene state as well as subsequent changes at specific times in the animation that cover a time period of the recorded experience.

Below is a representative Unreal Engine asset file as well as blueprints that are examples of the recorded experience file, retrieving the recorded experience file, defining the movements and attributes of objects, as well as playing back the recorded experience file.

[redacted]

[redacted]





16.3.1 Epic's Non-Infringement Contentions

Epic in their Third Amended Non-Infringement Chart for '605 (Exhibit G, page 1), states the following:

Epic denies that the Travis Scott Concert involved "retrieving a recorded experience file from the memory, the recorded experience file having been generated by saving an initial scene state and saving subsequent changes and respective times during a time period of the recorded experience." Every instance of the Travis Scott Concert, including every repeat performance, was generated using identical programming of an animation sequence involving hundreds of individually animated objects that players in each instance experienced independently from other instances. There was no initial "recorded experience" "during a time period" of which an initial scene state and subsequent changes were saved. Furthermore, the concert music was a sound recording implemented in the animation sequence; no music was recorded during the actual concert event. In addition, there was no "scene state" information saved that included avatars.

I disagree with Epic on their other assertions for the following reasons:

1) The animated concert sequences of the Travis Scott concerts were recorded in advance of the first concert for later playback in each of the five broadcasts. Mr.

Mark Imbriaco described that the triggering event starts a sequence "*refers to the preprogrammed – think of it as the choreography of the event that is predesigned that exists on the server and the client already."*[524] He goes on to say, "*When the live events start, they're not truly live. I mean, they are live, they're – but they're being driven by this preprogrammed sequence of movements that are designed using a technology called the sequencer.*"[525] Mr. Peter Axt described that the playlists for the concert was the same for each of the five performances.[526] The Epic Games technology of the Unreal Engine Sequencer, combined with the Fortnite environment provided, "A method of playing back a recorded experience in a virtual worlds system."

2) The concert sequences are stored in a recorded experience file for later playback (e.g., [redacted] (bates EPIC-SRC101).
3) As described in section Claim 1[iii] for the '605 patent, the attributes of objects in the scene are stored using the Unreal Engine Sequencer as the initial scene state. This includes data that describes the positions and poses of the Travis Scott mannequin at the time of the start of the scene (e.g., the initial scene state that happens at the beginning of the first scene.)[527]
4) Subsequent keyframes store attributes of objects in the scene at subsequent times and depict the values of attributes at a given specific time period. This includes data that describes the positions and poses of Travis Scott's mannequin at subsequent times. The Sequencer software will then interpolate the change in the attributes from the initial scene state to the first keyframe as well as between any two given keyframes in order to generating a smooth and continuous animation between sets of keyframes.[528]

[524] Deposition Testimony of Mr. Mark Imbriaco, 314:11-25
[525] Deposition Testimony of Mr. Mark Imbriaco, 315:1-5
[526] Deposition Testimony of Mr. Peter Axt, 298:13-18
[527] Deposition Testimony of Mr. Peter Axt, 164:7-10, 217:11-218:19
[528] https://docs.unrealengine.com/5.1/en-US/creating-animation-keyframes-in-unreal-engine/

5) When the recorded experience is played back during run-time, the Fortnite software that is running on the client device will render the initial scene state as well as all subsequent changes over time.[529]
6) The binary asset file contains the data that determines the movements and motions of the Travis Scott avatar is called [redacted] (bates EPIC-SRC101) and was played back in the same way for each of the five Travis Scott virtual performances that took place between April 23rd, 2020 and April 25th, 2020.[530] [531]
7) The music of Travis Scott was pre-recorded and was played back the same way for each of the five performances.[532] In this way, the music was also part of a recorded experience file, that resided in memory, and was reproduced in the same way for each of the five Travis Scott performances.[533]

16.4 Claim 1[iv]

"playing back the recorded experience file by rendering, for display by the at least one client device, objects of the initial scene state in the new instance, including one or more avatars, and rendering updates to the initial scene state based on the subsequent changes over the time period; and"

"playing back the recorded experience file by rendering, for display by the at least one client device,"

As described previously, the Unreal Engine Sequencer allows for the creation of animations (e.g., creating a recorded experience and then saving a recorded experience in a file (e.g., [redacted] bates EPIC-SRC101.)) The Fortnite environment that is running on

[529] Deposition Testimony of Mark Imbriaco, 118:8-120:16
[530] https://www.fortnite.com/news/astronomical
[531] EPIC00014958 – Cyclone-Jerky Run of Show
[532] Deposition Testimony of Mr. Peter Axt, 111:12-112:4, 294:24-298:18
[533] *Id.*

the client device is responsible playing back and rendering the recorded experience file by getting any needed the assets from server, loading it from disk, to memory, to the processor and GPU on the client device for rendering.[534] As discussed previously, the client device can be a PC, a Mac, or one of several gaming consoles including Sony PlayStation, Microsoft Xbox, Nintendo Switch, as well as Android devices and iOS devices.[535] [536]

"objects of the initial scene state in the new instance, including one or more avatars,"

Objects of the initial scene state simply mean objects that are present initially in the recorded experience file. An example of this could be the big round circular screen on a stage that is present when the Travis Scott concert begins.[537] This object is in the common space because there are three-dimensional blocking volumes that surround it so that the attendees' avatars are unable to get to it.[538] This object is in the recorded experience file because the projection on the screen is updated over time in the same way for each broadcast of the Travis Scott event.[539]

[534] Deposition Testimony of Mark Imbriaco, 118:8-18
[535] https://www.epicgames.com/help/en-US/fortnite-c5719335176219/technical-support-c5719372265755/what-platforms-or-devices-are-compatible-with-fortnite-a5720385202331
[536] https://www.fortnite.com/mobile
[537] https://www.youtube.com/watch?v=kjj3FwzRQms
[538] https://www.youtube.com/watch?v=kjj3FwzRQms (time 1:05 – 1:10)
[539] Deposition Testimony of Mr. Peter Axt, 298:13-18

Also, as mentioned previously, the Court has determined that the initial scene state does not have to include avatars, but may include avatars. In other words, the Court has determined that the initial scene state may include avatars but it is not required that it include avatars.[540]

and rendering updates to the initial scene state based on the subsequent changes over the time period;

As described above in section Claim 1[iii], the attributes of objects in the scene are stored using the Unreal Engine Sequencer as the initial scene state.[541] Subsequent keyframes store attributes of objects in the scene at subsequent times and depict the values of attributes at a given specific time period.[542] The Sequencer software will then interpolate the change in the attributes from the initial scene state to the first keyframe as well as between any two given keyframes in order to generating a smooth and continuous animation.[543] When the recorded experience is

[540] Courts Order Regarding Claims Constructions, page 13.
[541] Deposition Testimony of Peter Axt, 164:1-18, 217:6-221:3
[542] *Id.*
[543] https://docs.unrealengine.com/5.1/en-US/creating-animation-keyframes-in-unreal-engine/

played back at run-time, the Fortnite software running on the client device will render the initial scene state as well as all subsequent changes over time.[544]

The animated concert sequences of the Travis Scott concerts were recorded in advance for later playback in a virtual world.[545] [546] It should be noted that the Court has already determined that the initial scene state does not require the presence of avatars. Avatars may simply be included, but they are not required.[547] The recorded experience included objects (which could but was not required to include avatars) rendered over a time period. An example of this is the AstroWorld object in the Travis Scott concert. This large and detailed planet like object was animated and moved over time.[548]

Below is a representative Unreal Engine asset file and blueprints that are responsible for the recorded experience file, retrieving the recorded experience file, defining the movements and attributes of objects, and playing back the recorded experience file.

[redacted]

[redacted]

[redacted]

[544] Deposition Testimony of Mr. Mark Imbriaco, 118:8-18
[545] Deposition Testimony of Mr. Mark Imbriaco, 314:11–315:14
[546] EPIC-14958 Run of Show for the Travis Scott concert experience (aka 'Cyclone Jerky')
[547] Court Order Regarding Claims Construction, page 13
[548] Deposition Testimony of Mr. Peter Axt, 213:9-215:3.

[redacted]

16.4.1 Epic's Non-Infringement Contentions

Epic in their Third Amended Non-Infringement Chart for '605 (Exhibit G, page 2), states the following:

Epic denies that the Travis Scott Concert involved "playing back the recorded experience file by rendering, for display by the at least one client device, objects of the initial scene state in the new instance, including one or more avatars, and rendering updates to the initial scene state based on the subsequent changes over the time period." Every instance of the Travis Scott Concert, including every repeat performance, was generated using identical programming of an animation sequence involving hundreds of individually animated objects that players in each

instance experienced independently from other instances. At no point was a "recorded experience file" with "scene state" information saved during a "time period" of an earlier "recorded experience" used to play back a saved version of a previously instantiated concert performance. In addition, there was no "scene state" information saved that included avatars. Further, no instance of the Travis Scott Concert contained "one or more avatars" that had been present in any prior instance of the concert; Travis Scott himself was a preprogrammed nonplayer-character ("NPC"), not an avatar.

I disagree with Epic on their other assertions for the following reasons:

1) The animated concert sequences of the Travis Scott concerts were recorded in advance for later playback in a virtual world.[549] [550]
2) The concert sequences are stored in a recorded experience file for later playback (e.g., [redacted] (bates EPIC-SRC101).
3) As described in section Claim 1[iii] for the '605 patent, the attributes of objects in the scene are stored using the Unreal Engine Sequencer as the initial scene state. This includes data that describes the positions and poses of the Travis Scott mannequin at the initial time (e.g., the beginning of the first scene.)[551]
4) Subsequent keyframes store attributes of objects in the scene at subsequent times and depict the values of attributes at a given specific time period. This includes data that describes the positions and poses of the Travis Scott mannequin at subsequent times. The Sequencer software will then interpolate the change in the attributes from the initial scene state to the first keyframe as well as between any two given keyframes in order to generating a smooth and continuous animation.[552]
5) When the recorded experience is played back at run-time, the Fortnite software running on the client device will render the initial scene state as well as all subsequent changes over time.[553]

---

[549] Deposition Testimony of Mr. Mark Imbriaco, 314:11–315:15
[550] EPIC-14958 Run of Show for the Travis Scott concert experience (aka 'Cyclone Jerky')
[551] Deposition Testimony of Mr. Peter Axt, 164:1-18, 217:6-221:3
[552] https://docs.unrealengine.com/5.1/en-US/creating-animation-keyframes-in-unreal-engine/
[553] Deposition Testimony of Mr. Mark Imbriaco, 118:8-18

6) This binary file has the data that determines the movements and motions of the Travis Scott avatar that was played back in the same way for each of the five Travis Scott virtual performances that took place between April 23rd, 2020 and April 25th, 2020.[554] [555] [556]

7) The music of Travis Scott was pre-recorded and was played back the same way for each of the five performances.[557] In this way, the music was part of the recorded experience file, that resided in memory, and was reproduced in the same way for each of the five Travis Scott performances.[558]

8) Epic states, but the claim limitation does not require that "*At no point was a "recorded experience file" with "scene state" information saved during a "time period" of an earlier "recorded experience" used to play back a saved version of a previously instantiated concert performance.*

16.5 Claim 1[v]

"automatically transporting the one or more avatars to a different new instance of the scene, upon occurrence of threshold event, wherein the threshold event comprises when a maximum capacity of avatars has been reached in the new instance of the scene."

The matchmaking process as described above is responsible for placing players (or attendees in the case of the Travis Scott concerts) into one instance of the virtual world.[559] The matchmaking process ensures that the number of players in an instance of the scene is not greater than the maximum number of players allowed in that new instance.[560] The matchmaking process keeps track of how many players have been assigned to a given new instance and when the maximum number of players that are assigned to a given instance equals the maximum number

[554] https://www.fortnite.com/news/astronomical
[555] Deposition Testimony of Mr. Mark Imbriaco, 314:11–315:14
[556] EPIC-14958 Run of Show for the Travis Scott concert experience (aka 'Cyclone Jerky')
[557] Deposition Testimony of Mr. Peter Axt, 111:12-112:4, 294:24-298:18
[558] *Id.*
[559] Deposition Testimony of Mr. Peter Axt, 78:13-15
[560] Deposition Testimony of Mr. Peter Axt, 249:3-250:19

of players that are allowed for any given instance (e.g., the [redacted]).[561] When the number of players assigned to a given new instance equals the maximum number of players allowed for a given instance, that is threshold event that results in new additional players being put into a different new instance of the scene.

The first step is that when the user is in the Lobby, they can utilize the user interface to [redacted] request to matchmake [redacted]

[redacted]

Given that there was a maximum number of [redacted] per dedicated server session for the Travis Scott and Ariana Grande concert, players could only be assigned to a dedicated server session if there were fewer than [redacted].[570] It is not possible or a dedicated server to receive more than its configured maximum number of players.[571] [redacted]

---

[561] Deposition Testimony of Mr. Peter Axt, 249:3-250:19
[562] Deposition Transcript of Peter Axt at 236:6-21
[563] *Id.*
[564] *Id.*
[565] Deposition Transcript of Peter Axt at 237:14-19
[566] Deposition Transcript of Peter Axt at 237:11-14
[567] Deposition Transcript of Peter Axt at 239:2-12
[568] Deposition Transcript of Peter Axt at 241:17-242:11
[569] Deposition Transcript of Peter Axt at 243:9-244:21
[570] Deposition Transcript of Peter Axt at 245:23-246:22
[571] Deposition Transcript of Peter Axt at 247:5-21

[redacted][73]

Once a given dedicated server is full (e.g., [redacted]), the matchmaking is complete for that dedicated server.[574]

[redacted]

Mr. Mark Imbriaco explains that information about the world is stored using a replication graph. He describes the replication graph as being *"the piece of technology that is used to determine what changes in game state are relevant to each player at a point in time, and provide those updates to players, as needed, so that they have a reasonably accurate representation of the game state that's relevant to them"* and that the replication graph *"includes position of objects, characters, and other parts of game state."* [579] The game state information is delivered through the replication graph and this would include the four accused events.[580] Mr. Imbriaco

[572] Deposition Transcript of Peter Axt at 249:17-250:3
[573] Deposition Transcript of Peter Axt at 250:4-250:19
[574] Deposition Transcript of Peter Axt at 251:11-252:22
[575] Deposition Transcript of Peter Axt at 252:3-22
[576] *Id.*
[577] Deposition Transcript of Peter Axt at 255:2-18
[578] Deposition Transcript of Peter Axt at 256:5-24
[579] Deposition Testimony of Mark Imbriaco, 113:7-13, 113:17-18
[580] Deposition Testimony of Mark Imbriaco, 122:7-13

say that it is the server that distributes game-state changes to the clients.[581] [582] The Fortnite environment that is running on the client device is responsible playing back and rendering by getting any needed the assets from server (via the replication graph), and rendering it on the client device.[583] As discussed previously, the client device can be a PC, a Mac, or one of several gaming consoles including Sony PlayStation, Microsoft Xbox, Nintendo Switch, as well as Android devices and iOS devices.[584] [585]

In the way described above, Epic Games is able to play back information that is captured in the recorded experience file so that the client devices can render objects of the initial scene state that were present in the instances. The replication graph allows for sending the required information so that the clients can render updates to the initial scene state given the subsequent changes that are present in the replication graph over any time period.

It is the Epic technologies of Autoscaling, Dedicated Servers, Master Control Program and the Matchmaking Service that are described above that are responsible for making sure that fleets of new instances are available to host the millions of users that attended the Travis Scott concerts and that participants get put into new instances, but no more than the maximum number of participants allowed. This post from Epic's Fortnite's Twitter account states that there were [redacted] concurrent players that participated live in one of Travis Scott's concerts.[586] Given that there were [redacted] that were in each instance[587], this equates to approximately [redacted] concurrent instances that were needed to support the Travis Scott concert.

Below are representative source code files and blueprints that are responsible for limiting the number of avatars in any given instance and for automatically transporting avatars from the

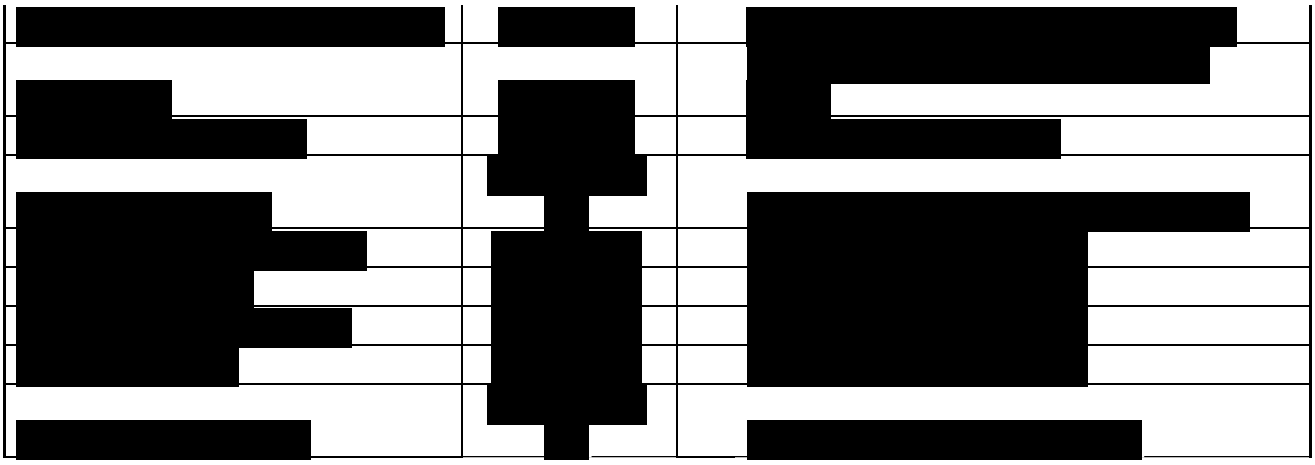
[581] Deposition Testimony of Mark Imbriaco, 126:11-15
[582] Deposition Testimony of Mark Imbriaco, 142:2-11
[583] Deposition Testimony of Mark Imbriaco, page 118
[584] https://www.epicgames.com/help/en-US/fortnite-c5719335176219/technical-support-c5719372265755/what-platforms-or-devices-are-compatible-with-fortnite-a5720385202331
[585] https://www.fortnite.com/mobile
[586] https://twitter.com/FortniteGame/status/1255524351376330752
[587] Deposition Testimony of Peter Axt, 246:12-14.

lobby to one of the new instances of the scene when the number of avatars in the instance that is currently being filled reaches a maximum capacity.

[redacted]

Name	Bates	Description

[588] https://docs.unrealengine.com/4.26/en-US/API/Plugins/OnlineSubsystemUtils/APartyBeaconHost/



16.5.1 Epic's Non-Infringement Contentions

Epic in their Third Amended Non-Infringement Chart for '605 (Exhibit G, pages 2 - 3), states the following:

Epic denies that the Travis Scott Concert involved "automatically transporting the one or more avatars to a different new instance of the scene, upon occurrence of a threshold event, wherein the threshold event comprises when a maximum capacity of avatars has been reached in the new instance of the scene."

Epic denies that the Travis Scott Concert involved "automatically transporting the one or more avatars to a different new instance of the scene." No instance of the Travis Scott Concert contained "one or more avatars" that had been present in any prior instance of the concert and rendered from a recorded experience file, nor were such avatars transported to a new instance of a scene. Travis Scott himself was a preprogrammed NPC, not an avatar. Identical copies of the Travis Scott NPC were created in every instance; the Travis Scott NPC was not "transported" between instances. Further, player avatars could not travel between instances of the Travis Scott Concert, when a maximum capacity has been reached or otherwise; each player was assigned to an instance by the Matchmaking Service ("MMS") before the instance and the player's avatar were generated and player avatars remained in that instance for the entire performance.

Epic further denies that avatars were transported "upon occurrence of a threshold event, wherein the threshold event comprises when a maximum capacity of avatars has been reached in the new instance of the scene." The instances of the game environment hosting the Travis Scott Concert were generated using MMS, a service that detected and matched a pool of available players according to different attributes of the players, the client devices, and their connections, before the game environment, and the player's avatars, were generated; a player avatar could not enter an instance of the Travis Scott Concert without first being matched with a pool of other

players. Moreover, the MMS would match and populate instances at numbers below the maximum threshold of avatars for each instance.

I disagree with Epic on their other assertions for the following reasons:

1) The matchmaking process as described above is responsible for placing players (or attendees in the case of the Travis Scott concerts) into one instance of the virtual world.[589] The matchmaking process ensures that the number of players in an instance of the scene is not greater than the maximum number of players allowed in that new instance.[590]
2) The matchmaking process keeps track of how many players have been assigned to a given new instance and when the maximum number of players that are assigned to a given instance equals the maximum number of players that are allowed for any given instance (e.g., [redacted]).[591] When the number of players assigned to a given new instance equals the maximum number of players allowed for a given instance, that is threshold event that results in new additional players being put into a different new instance of the scene.
3) Avatars were automatically transported from the Lobby to the first scene once matchmaking was complete. The attendees had the same avatar while in the Lobby as they did when they were in the first scene. The attendees were automatically transported out of the Lobby and into a new instance upon the occurrence of a threshold event. The threshold event is when a previous instance is full. In the case of Travis Scott, there was a maximum of [redacted] that were supported by each instance.[592]
4) Epic states that "player avatars could not travel between instances of the Travis Scott Concert." The claim limitation does not require that avatars be able to travel between instances of the Travis Scott Concert.

---

[589] Deposition Testimony of Mr. Peter Axt, 78:12-15
[590] Deposition Testimony of Mr. Peter Axt, pages 249:3-250:19
[591] *Id.*
[592] Deposition Testimony of Peter Axt, page 245:23-246:22.

16.6 Claim 2

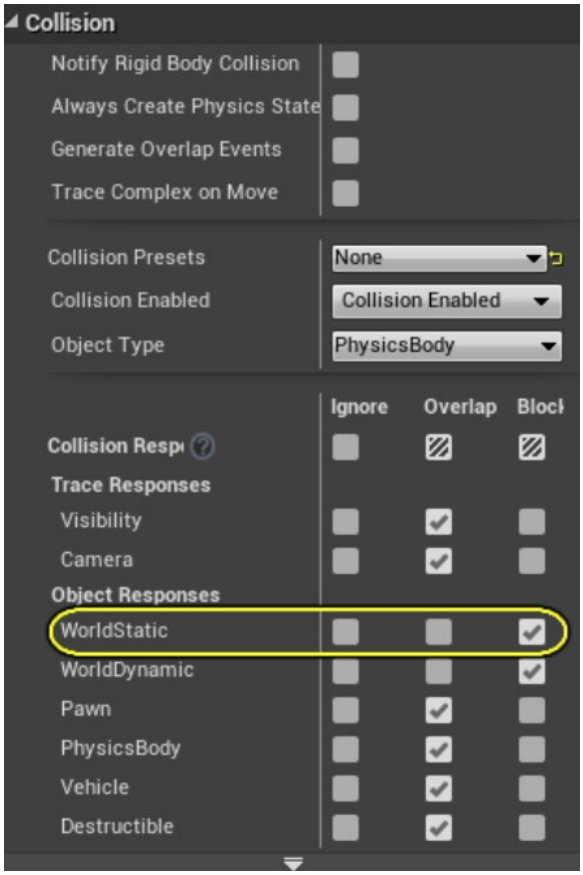
"The method of claim 1, wherein movement within the new instance by the one or more avatars associated with at least one client device is limited by objects of the recorded experience."

The way that Fortnite is designed (as well as the design of the Travis Scott concerts), each individual player is associated with only one client device that they use to interface with Fortnite and the concert. In addition, each individual player is associated with only one avatar. Also, as discussed previously in this report, Epic uses three-dimensional blocking volumes in order to limit movement of avatars.[593]

The Unreal Engine documentation titled Collision Overview describes in detail how blocking works within Unreal Engine.[594] The document states that "Blocking will naturally occur between two (or more) Actors set to Block."

This Unreal Engine dialog box below allows the user of Unreal Engine to set the attributes of a given object to Block, that is what is desired. Additionally, Unreal Engine supports the automatic generation of hit events when a collision takes place with events such as ReceiveHit or OnComponentHit. In this way, the designers and developers of the Travis Scott concerts created blockers (that were included into the sequence and were therefore part of the recorded experience file) that limited the movement of avatars within the instance.

[593] Deposition Testimony of Peter Axt, 160:2-161:20
[594] EPIC-20389

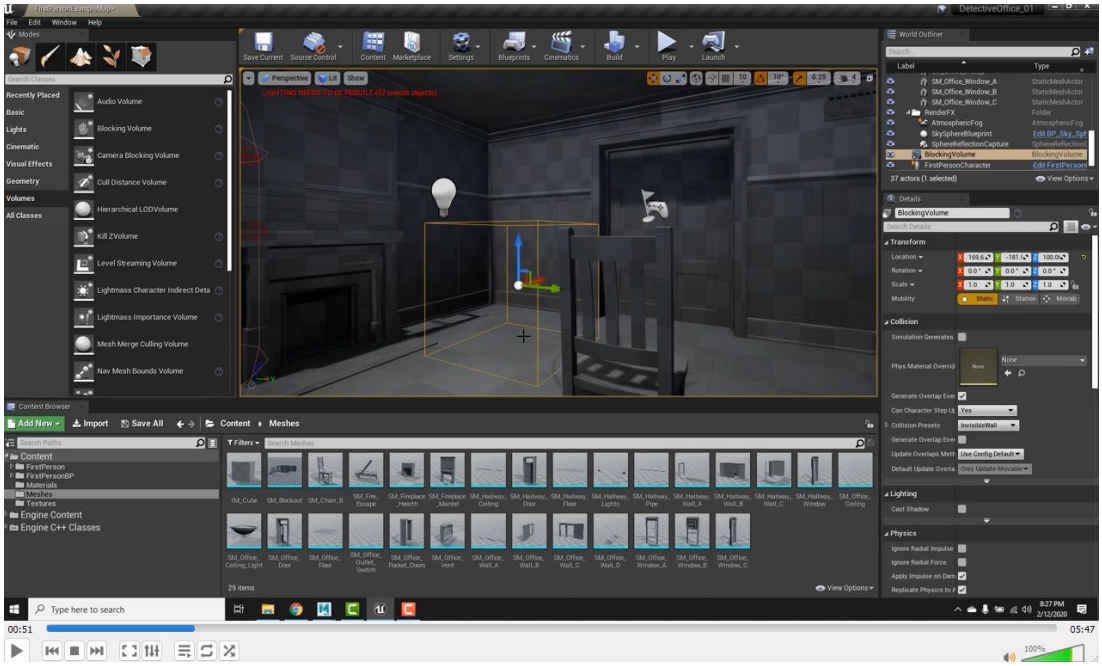


In the same way that was described above for the infringement analysis of the Travis Scott concerts on '071 patent, participant avatars could move about within the concert venue, but only within defined limits of that venue.[595] Blocking volumes defined an area or stage (e.g., the common area) that represented an area that the participant avatars could not move to. The blocking volumes are invisible three-dimensional objects that set limits on where participant avatars can travel.[596]

[595]

[596] EPIC-00020291

This video that is in this footnote shows how users can setup blocking volumes in Unreal Engine.[597] If one looks at time 0:25 – 1:30, the instructor shows how to utilize three-dimensional blocking volumes can be setup to limit the motion of players within the scene.



Here is another video that shows how the blocking volumes prevented user avatars from entering the common space. https://www.youtube.com/watch?v=kjj3FwzRQms If one looks at times 1:05 to 1:11, one will see how the blocking volumes prevented user avatars from entering the stage (e.g., the common space). In looking at times 7:20 to 7:35, one can see how the blocking volumes prevented user avatars from being too close the mannequin of Travis Scott when he was under water.

[597] EPIC-00020405.mp4

Below are representative source code files and blueprints that are responsible for both moving the avatars that are associated with players (and client devices), limiting the movement of avatars that are controlled by players, and the blueprint for the recorded experience file that contains blocking volumes.

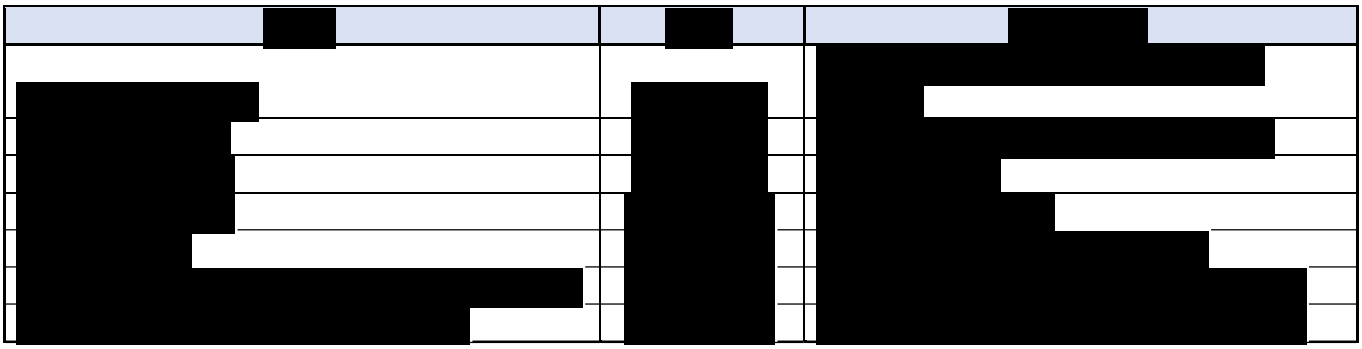
[redacted]

[redacted]

[redacted]

[redacted]

[redacted]



16.6.1 Epic's Non-Infringement Contentions

Epic in their Third Amended Non-Infringement Chart for '605 (Exhibit G, page 3), states the following:

Epic denies that the Travis Scott Concert included that "movement within the new instance by the one or more avatars . . . is limited by objects of the recorded experience." No objects of a recorded experience were present in any instance of the Travis Scott Concert. No instance of the Travis Scott Concert contained "one or more avatars" that had been present in any prior instance of the concert and rendered from a recorded experience file. The Travis Scott NPC's movement was not limited by any object. Further, player avatars' movements were limited by objects and map settings in a newly generated instance of the Travis Scott Concert, not by objects from a recorded experience.

I disagree with Epic on their other assertions for the following reasons:

1) As discussed previously in this report, Epic uses three-dimensional blocking volumes in order to limit movement of avatars. The Unreal Engine documentation titled Collision Overview describes in detail how blocking works within Unreal Engine.[598] The document states that "Blocking will naturally occur between two (or more) Actors set to Block." In this way, the designers and developers of the Travis Scott concerts created blockers (that were included into the sequence and were therefore part of the recorded experience file) that limited the movement of avatars within the instance.[599]
2) The three-dimensional blocking volumes (e.g., [redacted]) were part of the recorded experience that is captured in the file [redacted].[601]

16.7 Claim 5

"The method of claim 1, wherein the recorded experience file is not modifiable by events occurring during playback of the recorded experience."

As discussed previously, the Travis Scott event took place five times from Aug 23rd, 2020 through Aug 25th, 2020. The recorded experience file was the same in all Travis Scott events.[602]

---

[598] EPIC-20389
[599] Deposition Testimony of Peter Axt, 158:2-160:22
[600] EPIC-SRC159
[601] EPIC-SRC101
[602] Deposition Testimony of Mr. Peter Axt, 298:5-18

While the users (participant avatars) could perform new actions in the virtual world, but those new actions would not modify the recorded file itself.

In reviewing the Epic source code in February and March of 2023, I found no evidence that the recorded experience file was modifiable by events occurring during playback of the recoded experience.

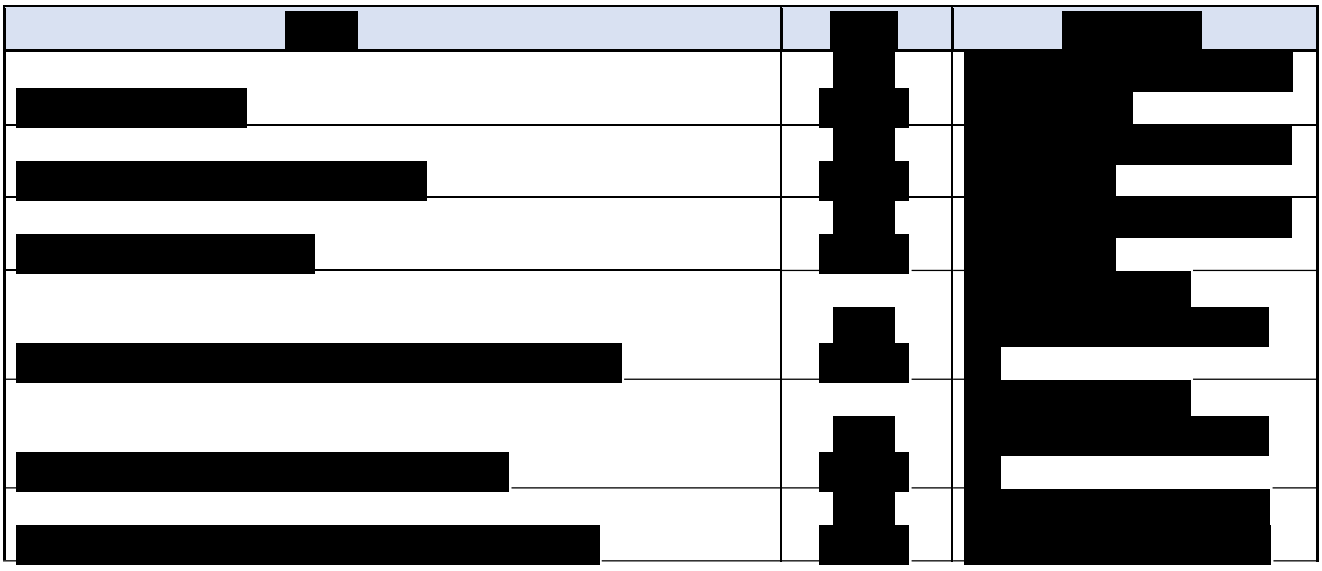
In addition, the music of Travis Scott, which is part of the recorded experience was also not modifiable by events during the playback of the concerts. The music was pre-recorded and the visuals were created to coincide with the music. The music and the visuals were presented the same way each of the five times that the Travis Scott event was presented.[603] The website Vox.com reported, "All of these visuals are timed to the music, so Scott isn't actually playing his songs live."[604] This shows that the recorded experience file that contained the music was not modified during playback of the recorded experience.

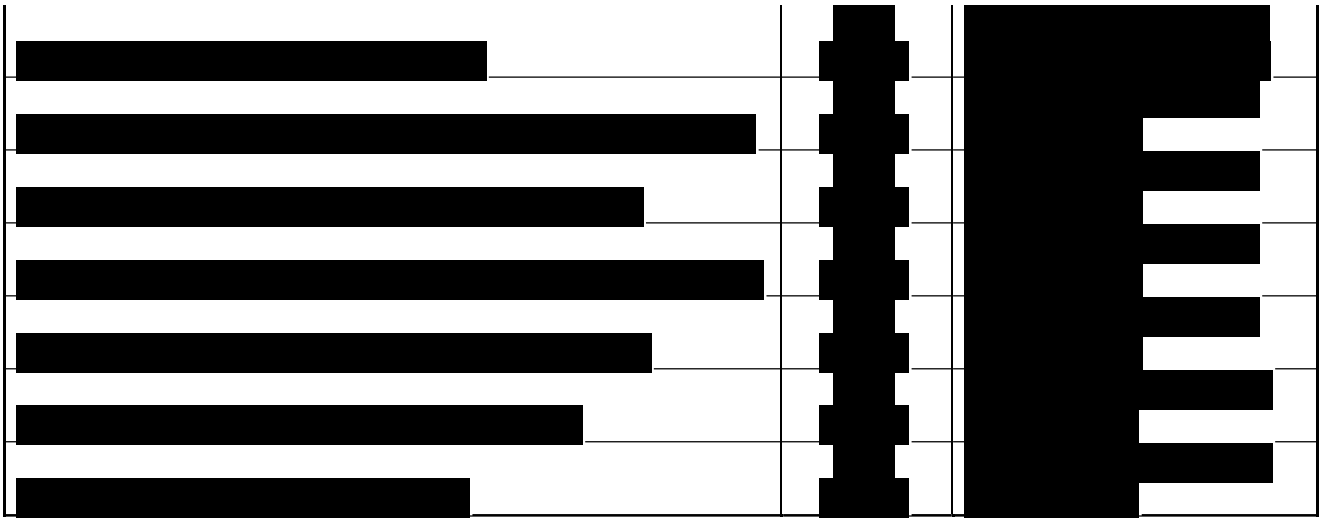
Below is the relevant recorded experience file, source code files, and blueprints that define the recorded experience file and control the playback of the recorded experience during run-time of the Travis Scott concerts.

[redacted]

[603] Deposition Testimony of Mr. Peter Axt, 298:5-18
[604] https://www.vox.com/culture/2020/4/24/21235196/travis-scott-fortnite-concert-livestream-the-scotts-music-video

[redacted]





16.7.1 Epic's Non-Infringement Contentions

Epic in their Third Amended Non-Infringement Chart for '605 (Exhibit G, pages 3 - 4), states the following:

Epic denies that the Travis Scott Concert included that "the recorded experience file is not modifiable by events occurring during playback of the recorded experience." The Travis Scott Concert did not involve playing back a recorded experience file; rather, it was an interactive, pre-programmed sequence animated in real-time. Accordingly, there was no recorded experience file that was capable or incapable of being modified.

I disagree with Epic on their other assertions for the following reasons:

1) The Travis Scott concert did playback a recorded experience file. The recorded experience file was [redacted] (bates EPIC-SRC101.) This file was created using the Unreal Engine Sequencer using keyframe animation.
2) As described in this report, the movements of various elements in the Travis Scott concerts were preprogrammed using the keyframe animation in the Sequencer of Unreal Engine.[605] These recordings were played back in the same way for each of the five Travis Scott concerts that took place between April 23rd, 2020 and April 25th,

[605] https://docs.unrealengine.com/5.1/en-US/unreal-engine-sequencer-movie-tool-overview/

2020.[606] The Unreal engine Sequencer created a recorded experience file [redacted] (bates EPIC-SRC101) that was used to playback the animation of the mannequin of Travis Scott in the same way each time the concert was broadcast.[607]

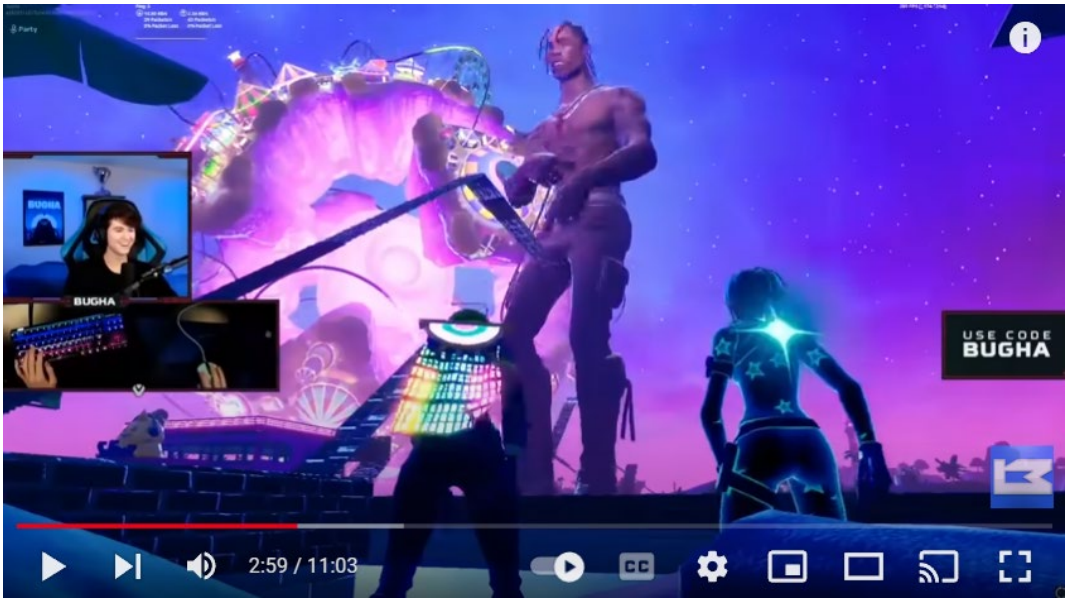
3) In reviewing the Epic source code in February and March of 2023, I found no evidence that the recorded experience file was modifiable by events occurring during playback of the recoded experience.

16.8 Claim 8

"The method of claim 1, wherein the new instance of the scene is three-dimensional."

Epic admits that the instances of the Travis Scott Concert were three-dimensional.[608]

As one can see from the videos of the Travis Scott concerts, the environment was three-dimensional.[609] Another video of the same event can be found here.[610]



606 https://www.fortnite.com/news/astronomical
607 607 Deposition Testimony of Mr. Peter Axt, 298:5-18
608 Third Amended Non-Infringement Chart of '605 Patent – Exhibit G – Page 4
609 https://www.youtube.com/watch?v=kjj3FwzRQms
610 https://www.youtube.com/watch?v=gYP9Katzh0c

16.8.1 Epic's Non-Infringement Contentions

Epic in their Third Amended Non-Infringement Chart for '605 (Exhibit G, page 4), states the following:

Epic admits that the instances of the Travis Scott Concert were three-dimensional.

I agree with Epic that, "*that the instances of the Travis Scott Concert were three-dimensional"* and the videos from the events prove that as well.[611]

17 INFRINGEMENT OF THE '605 PATENT BY ARIANA GRANDE

17.1 Claim 1[i]

"A method of playing back a recorded experience in a virtual worlds system, comprising:"

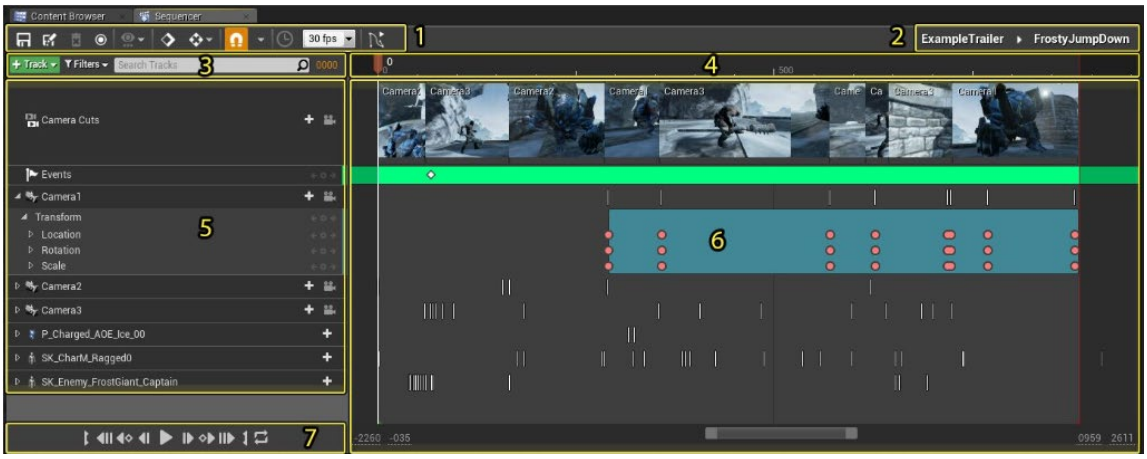
As an initial matter, in the Court's claim construction order, the Court said, "After considering the briefing of the parties and arguments at the hearing, the Court agrees with Utherverse and declines to rule that the preambles are limiting."[612]

Playing back a recorded experience implicitly requires that the experience is first recorded. As described above, Epic Games has created the Sequencer tool that allows for the creation of sequences that are stored in a recorded experience file. In Unreal Engine, sequences can be created and played back with the use of the Sequencer, which is a powerful cinematic editing tool that enables the creation of real-time, animations, in-engine cutscenes, cinematics, and scripted events.[613] Sequencer allows for the creation of complex sequences by combining

[611] https://www.youtube.com/watch?v=kjj3FwzRQms
[612] Order regarding Claims Construction, page 15
[613] https://docs.unrealengine.com/4.26/en-US/AnimatingObjects/Sequencer/Overview/

various elements such as animations, audio, camera shots, and lighting in a linear or non-linear fashion.[614]

The image below, from the Unreal Engine documentation, shows the various parts of the Sequencer Editor including; 1) Tool Bar, 2) Sequence Breadcrumbs, 3) Add/Filter Tracks, 4) Timeline, 5) Tree View, 6) Tracks Area, and 7) Playback Controls.[615]

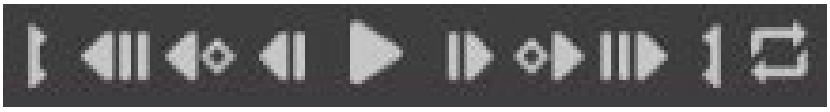


This section of the Epic Games documention on the Unreal Engine Sequencer describes, among other things, how the Sequencer Editor is used to both create animations and playback animations. The documentation shows using the CurveEditor to fine tune animation keys that are placed in the Tracks Area as well as adjusting properites of objects in the tracks area of the Sequencer such as start offset, end offset, play rate, etc.

In addition to the recording of animations, the Unreal Engine Sequencer supports playback of animations as well (see section 7 in the image above.) The Playback Controls section of the documentation discusses multiple features to assist animators in playing back the recorded experience.[616] These features include the following commands and controls.

---

[614] https://www.youtube.com/watch?v=D3trMpkxAFk
[615] https://docs.unrealengine.com/4.26/en-US/AnimatingObjects/Sequencer/ReferenceEditor/
[616] https://docs.unrealengine.com/4.26/en-US/AnimatingObjects/Sequencer/ReferenceEditor/



Command	Description
	Set the playback start position to the current position indicated by the time marker.
	Jump to the playback start position.
	Jump to the previous key in the selected track(s).
	Jump to the previous frame.
	Play or Pause the Level Sequence from the position of the time marker.
	Jump forward a frame.
	Jump to the next key in the selected track(s).
	Jump to the playback end position.
	Set the playback end position to the current position indicated by the time marker.
/	Toggles between looping the Level Sequence during playback.

In addition, the Sequencer supports the ability to manipulate the animation playback rate through the use of a Play Rate Track.[617] Through the use of the Unreal Engine Sequencer, Epic Games has the ability to both record animations at design time, and then playback the animations at either design time and/or at run-time.

[617] https://docs.unrealengine.com/4.26/en-US/AnimatingObjects/Sequencer/HowTo/TracksPlayRate/

Unreal Engine assets, including animation sequence data that represent the recorded experience file are stored in binary files that end with the extension ".uasset".[618] This is the native Unreal Engine file format for Unreal Engine assets. For example, the file [redacted] (bates EPIC-SRC3), is an asset file that holds the animation of the mannequin of Ariana Grande.

This binary file has the data that determines the movements and motions of the Ariana Grande avatar that was played back in the same way for each of the five Ariana Grande virtual performances that took place between Aug 6th, 2021 and Aug 8th, 2021.[619] [620] [621] It should be noted that the Epic Games technologies, which includes the Sequencer of the Unreal Engine, allowed for both the creation and playing back of the animation of the mannequin of Ariana Grande.

The animation of the mannequin of Ariana Grande was a recorded experience using the Unreal Engine Sequencer and it was encapsulated in one or more files (e.g., [redacted]), and it was played back in the same way for each of the five performances.[622] Mr. Mark Imbriaco described a triggering event that started the sequence "*refers to the preprogrammed – think of it as the choreography of the event that is predesigned that exists on the server and the client already.*"[623] He goes on to say, "*When the live events start, they're not truly live. I mean, they are live, they're – but they're being driven by this preprogrammed sequence of movements that are designed using a technology called the sequencer.*"[624] The Epic Games technology of the Unreal Engine Sequencer, combined with the Fortnite environment provided, "A method of playing back a recorded experience in a virtual worlds system."[625]

---

[618] Deposition Testimony of Peter Axt, 221:8-13
[619] https://www.epicgames.com/site/en-US/news/ariana-grande-steps-into-the-metaverse-as-the-headliner-for-fortnites-rift-tour
[620] EPIC00014957 – Buffet Run of Show
[621] Deposition Testimony of Mr. Mark Imbriaco, 314:11–315:14
[622] Deposition Testimony of Mr. Peter Axt, 298:21-299:3
[623] Deposition Testimony of Mr. Mark Imbriaco, 314:11-25
[624] Deposition Testimony of Mr. Mark Imbriaco, 315:1-5
[625] 605 Patent, Claim 1[i]

In addition to the recorded experience file of "[redacted]" (bates EPIC-SRC3), below are representative blueprint files that are responsible for playing back the recorded experience file. [redacted]

[626] Deposition Testimony of Peter Axt, 221:12-13
[627] Deposition Testimony of Peter Axt, 224:6-7

Name	Bates	Description
[redacted]	[redacted]	[redacted]

17.1.1 Epic's Non-Infringement Contentions

Epic in their Third Amended Non-Infringement Chart for '605 (Exhibit K, page 1), states the following:

To the extent the preamble is limiting, Epic denies that the Ariana Grande Concert involved a "method of playing back a recorded experience in a virtual worlds system." The Ariana Grande Concert was not a "playing back" of a "recorded experience," but rather an interactive, preprogrammed sequence animated in real-time.

I disagree with Epic on their other assertions for the following reasons:

1) Playing back a recorded experience implicitly requires that the experience is first recorded. As described above, Epic Games has created the Sequencer tool that allows for the creation of sequences that are stored in a recorded experience file. In Unreal Engine, sequences can be created and played back with the use of the Sequencer, which is a powerful cinematic editing tool that enables the creation of real-time, animations, in-engine cutscenes, cinematics, and scripted events.[628] Sequencer allows for the creation of

[628] https://docs.unrealengine.com/4.26/en-US/AnimatingObjects/Sequencer/Overview/

complex sequences by combining various elements such as animations, audio, camera shots, and lighting in a linear or non-linear fashion.[629]

2) This section of the Epic Games documention on the Unreal Engine Sequencer describes, among other things, how the Sequencer Editor is used to both create animations and playback animations.[630] The Playback Controls section of the documentation discusses multiple features to assist animators in playing back the recorded experience.[631]
3) As described in this report, the movements of various elements in the Ariana Grande concerts were preprogrammed using the keyframe animation in the Sequencer of Unreal Engine. These recordings were played back in the same way for each of the five Ariana Grande concerts that took place between August 6th, 2021 and August 8th, 2021.[632] [633] The Unreal engine Sequencer created a recorded experience file [redacted] (bates EPIC-SRC3) that was used to playback the animation of the jeweled hammer and avatar of Ariana Grande in the same way each time the concert was broadcast.[634] [635]

17.2 Claim 1[ii]

"instantiating, using one or more processors of a server, a new instance of a scene, the new instance being defined by data stored in memory, at least one client device displaying and participating in the new instance;"

First, it should be noted that Epic admits infringement of this claim limitation. In their Third Amended Non-Infringement Contentions, Epic states, "*Epic admits that the Ariana*

[629] https://www.youtube.com/watch?v=D3trMpkxAFk
[630] https://docs.unrealengine.com/4.26/en-US/AnimatingObjects/Sequencer/ReferenceEditor/
[631] https://docs.unrealengine.com/4.26/en-US/AnimatingObjects/Sequencer/ReferenceEditor/
[632] https://www.epicgames.com/site/en-US/news/ariana-grande-steps-into-the-metaverse-as-the-headliner-for-fortnites-rift-tour
[633] Deposition Testimony of Mr. Peter Axt, 298:21-299:3
[634] EPIC-14957 – Buffet - Run of Show
[635] Deposition Testimony of Mr. Mark Imbriaco, 314:11–315:14

Grande Concert involved creating new instances of a scene, and that players participated in the instances using client devices." [636]

Multiple Epic technologies were utilized to accomplish this, as discussed below.

"instantiating, using one or more processors of a server, a new instance of a scene, the new instance being defined by data stored in memory"

The plurality of parallel dimensions in a computer memory were provided by provisioning fleets of virtual machines using EC2 and GCE instances built from a single VM image.[637] A virtual machine has all of the same capabilities of a real computer including processing, memory, storage, networking, etc.[638] [639] The section above on Autoscaling, Dedicated Servers, Master Control Program (MCP), and Matchmaking Service (MMS) above explains how Epic Games utilizes multiple servers to ensure that the events run smoothly, with minimal latency in order to account for variable and fluctuating user demand.

Each new instance of a server virtual machine gets a copy of a master virtual image. This is described by Epic in describing how they use fleets of dedicated servers that are 100% identical to one another. Fleets are resized by adding and removing compute instances (EC2 or GCE instances built from a single VM Image) from the fleet.[640] In this way, each of the virtual machine instances are duplicates of one another and Epic's own internal documents say that they are "100% identical to one another."[641] Mr. Imbriaco also said that the dedicated server processes all come from the same installation package.[642] He also said that fleets are resized by adding and removing instances built from the same virtual computer image.[643] He also said that it is the Fortnite developers that create the installation packages.[644] He also said that the VM

---

[636] Third Amended Non-Infringement Chart of '605 Patent – Exhibit K – Page 1
[637] EPIC-00020439 (page 2)
[638] Deposition Testimony of Mark Imbriaco, 212:8-12
[639] Deposition Testimony of Peter Axt, 23:23-24:6
[640] EPIC-00020439
[641] EPIC-00020439
[642] Deposition Transcript of Mark Imbriaco, 203:14-204:11
[643] Deposition Transcript of Mark Imbriaco, 205:3-207:25
[644] Deposition Transcript of Mark Imbriaco, 117:2-18

image has the Fortnite version is stored in the build source that is created by Epic Games developers.[645]

On the server side, each instance is its own virtual machine with its own virtual memory that is used to host a small subset of participants. Each virtual machine has its own dedicated virtual memory and each virtual machine is located within one physical server that has its own physical memory.[646] [647] [648] On the client side, each client device also has its own memory as well. As discussed previously, the client device can be a PC, a Mac, or one of several gaming consoles including Sony PlayStation, Microsoft Xbox, Nintendo Switch, as well as Android devices and iOS devices.[649] [650]

"at least one client device displaying and participating in the new instance;"

On the client side, the various technologies of the Master Control Program (MCP), and Matchmaking Service (MMS) ensure that the client device gets connected to a specific dedicated server that has capacity and only needs to handle the state of the virtual world for the much smaller subset of attendees that are associated with that dedicated server.[651]

The first step is that when the user is in the Lobby, they can utilize the user interface to [redacted] request to matchmake [redacted]

[645] Deposition Transcript of Mark Imbriaco, 262:1-20
[646] Deposition Testimony of Mark Imbriaco, 212:8-12
[647] Deposition Testimony of Peter Axt, 23:23-24:6
[648] https://aws.amazon.com/what-is/virtualization/
[649] https://www.epicgames.com/help/en-US/fortnite-c5719335176219/technical-support-c5719372265755/what-platforms-or-devices-are-compatible-with-fortnite-a5720385202331
[650] https://www.fortnite.com/mobile
[651] Deposition Testimony of Peter Axt at 34
[652] Deposition Transcript of Peter Axt at 236:6-21
[653] *Id.*
[654] *Id.*

[redacted]

Given that there was a maximum number of [redacted] per dedicated server session for the Travis Scott and Ariana Grande concert, players could only be assigned to a dedicated server session if there were fewer than [redacted].[660] It is not possible or a dedicated server to receive more than its configured maximum number of players.[661] [redacted][663] Once a given dedicated server is full (e.g., [redacted]), the matchmaking is complete for that dedicated server.[664]

[redacted]

[655] Deposition Transcript of Peter Axt at 237:14-19
[656] Deposition Transcript of Peter Axt at 237:11-14
[657] Deposition Transcript of Peter Axt at 239:2-12
[658] Deposition Transcript of Peter Axt at 241:17-242:11
[659] Deposition Transcript of Peter Axt at 243:9-244:21
[660] Deposition Transcript of Peter Axt at 245:23-246:22
[661] Deposition Transcript of Peter Axt at 247:5-21
[662] Deposition Transcript of Peter Axt at 249:17-250:3
[663] Deposition Transcript of Peter Axt at 250:4-250:19
[664] Deposition Transcript of Peter Axt at 251:11-252:22
[665] Deposition Transcript of Peter Axt at 252:3-22

[redacted]

Mr. Mark Imbriaco explains that information about the world is stored using a replication graph. He describes the replication graph as being *"the piece of technology that is used to determine what changes in game state are relevant to each player at a point in time, and provide those updates to players, as needed, so that they have a reasonably accurate representation of the game state that's relevant to them"* and that the replication graph *"includes position of objects, characters, and other parts of game state."* [669] The game state information is delivered through the replication graph and this would include the four accused events.[670] Mr. Imbriaco say that it is the server that distributes game-state changes to the clients.[671] [672] The Fortnite environment that is running on the client device is responsible playing back and rendering by getting any needed the assets from server (via the replication graph), and rendering it on the client device.[673] As discussed previously, the client device can be a PC, a Mac, or one of several gaming consoles including Sony PlayStation, Microsoft Xbox, Nintendo Switch, as well as Android devices and iOS devices.[674] [675]

---

[666] *Id.*
[667] Deposition Transcript of Peter Axt at 255:2-18
[668] Deposition Transcript of Peter Axt at 256:5-24
[669] Deposition Testimony of Mark Imbriaco, 113:7-13, 113:17-18
[670] Deposition Testimony of Mark Imbriaco, 122:7-13
[671] Deposition Testimony of Mark Imbriaco, 126:11-15
[672] Deposition Testimony of Mark Imbriaco, 142:2-11
[673] Deposition Testimony of Mark Imbriaco, page 118
[674] https://www.epicgames.com/help/en-US/fortnite-c5719335176219/technical-support-c5719372265755/what-platforms-or-devices-are-compatible-with-fortnite-a5720385202331
[675] https://www.fortnite.com/mobile

In the way described above, Epic Games creates and populates new instances so that the multiple client devices are able to connect to these new instances so they can participate in these new instances, get updates from these instances, and display the information that comes from each of the new instances.

Below are several representative source code files and blueprints that were used in assigning participants to new instances (e.g., [redacted]) and reservations during the matchmaking process (e.g., [redacted]) for the Ariana Grande concert.

[redacted]

[redacted]

[676] Deposition Testimony of Peter Axt, 246:15-17
[677] Deposition Testimony of Peter Axt, 311:18-312:4

17.2.1 Epic's Non-Infringement Contentions

Epic in their Third Amended Non-Infringement Chart for '605 (Exhibit K, page 1), states the following:

Epic admits that the Ariana Grande Concert involved creating new instances of a scene, and that players participated in the instances using client devices.

I agree with Epic that, "*the Ariana Grande Concert involved creating new instances of a scene, and that players participated in the instances using client devices.*"

As Epic already agrees that it infringes Claim 1[ii] of the '605 patent, I will keep this analysis short. Previously in this report, I describe how Epic uses technologies of Autoscaling,

Dedicated Servers, Master Control Program (MCP), and Matchmaking Service (MMS) to create a fleet of servers and then connect client devices to a specific server. I also describe how the virtual machines that are created have virtual memory that are assigned to them and these virtual machines reside on physical servers that have physical memory.[678] [679] [680]

17.3 Claim 1[iii]

"retrieving a recorded experience file from the memory, the recorded experience file having been generated by saving an initial scene state and saving subsequent changes and respective times during a time period of the recorded experience;"

"retrieving a recorded experience file from the memory"

As described above, the Unreal Engine Sequencer has the ability to create recorded experiences and save them in a file (e.g., the recorded experience file.) In the case of the Ariana Grande virtual concerts, the file "[redacted]" (bates EPIC-SRC3) is the Unreal Engine proprietary binary asset file that stores animation data for the mannequin of Ariana Grande, so that the mannequin moves and gestures in the same way each time that the recorded experience file is played back.[681]

In order to utilize the [redacted] file, when loading the Unreal Engine project, Epic software would retrieve this file from non-volatile memory (e.g., SSD or spinning hard disk) and copy it into volatile memory (e.g., RAM.) Once the file was in volatile memory (e.g., RAM), the computer processor could retrieve this file from RAM for use. Both the retrieving of the recorded experience file from non-volatile memory as well as the retrieving the file from RAM would constitute "retrieving a recorded experience file from the memory."

---

[678] Deposition Testimony of Mark Imbriaco, 212:8-12
[679] Deposition Testimony of Peter Axt, 23:17-24:6
[680] https://aws.amazon.com/what-is/virtualization/
[681] Deposition Testimony of Mr. Peter Axt, 298:21-299:3

"the recorded experience file having been generated by saving an initial scene state"

With regard to "the recorded experience file having been generated by saving an initial scene state," the Unreal Engine Sequencer allows for the saving of an initial scene state. A central component of the Unreal Engine Sequencer is the timeline. The timeline supports defining an initial scene state.[682]

The timeline supports both a green start marker and a red end marker. The start time is typically at time zero. This represents the initial state of the scene, or "the initial scene state." The user of the Unreal Engine Sequencer and timeline is free to set and save any initial attributes of the scene that they like at the beginning of the timeline. Examples of attributes that can be set for the initial scene might include the X, Y, and Z position of objects in the scene, the X, Y, and Z rotation of objects in the scene, the scaling of objects in the scene, the color of objects in the scene, various shader and lighting attributes of objects in the scene, data associated with a specific poses or positions of an avatar, etc.

"and saving subsequent changes and respective times during a time period of the recorded experience;"

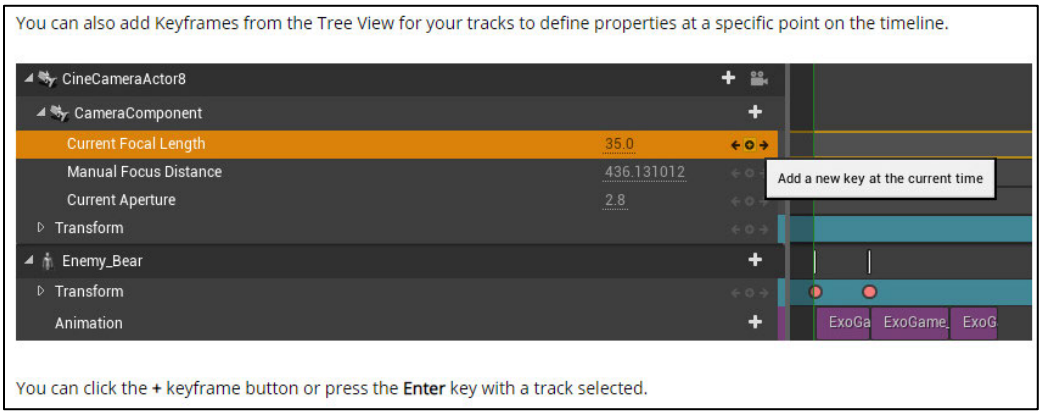
The Unreal Engine Sequencer supports keyframe animation. Keyframe animation in Unreal Engine refers to the process of creating animations by defining specific poses or transformations for 3D objects at certain points in time, known as keyframes. These keyframes are then interpolated by the engine to generate all of the in-between frames, thus creating a smooth animation sequence.

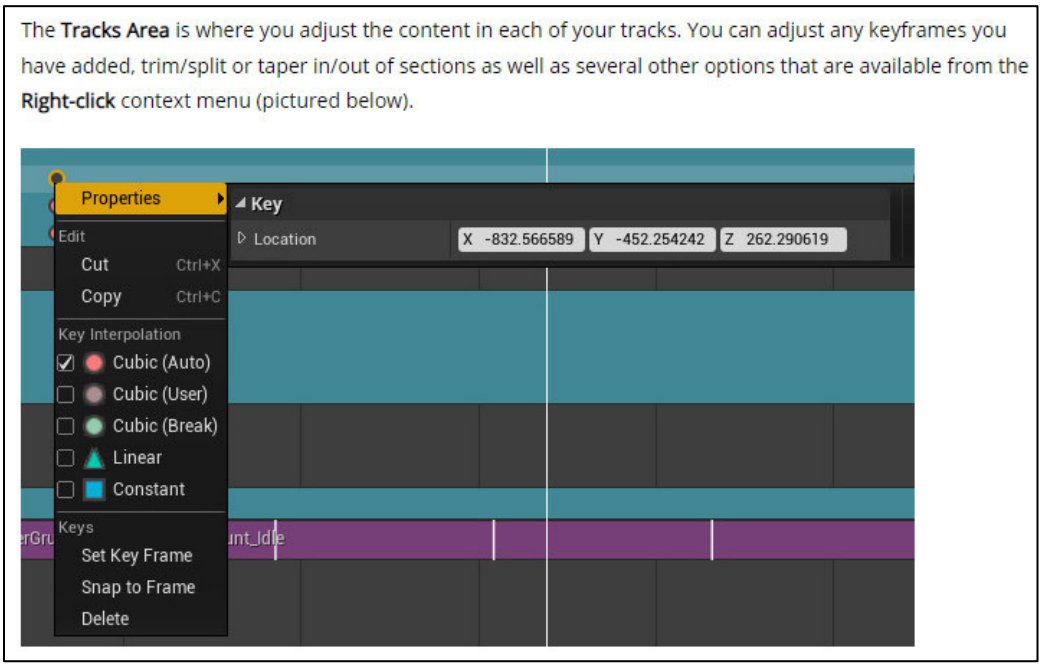
The Unreal Engine Sequencer allows users to create animations by setting keyframes for various properties like position, rotation, and scale of 3D objects, as well as cameras, lights, attributes of specific poses and positions of avatars, and other elements at specific points in time in the animation timeline. The properties that are set at the beginning of the timeline specify the

[682] https://docs.unrealengine.com/4.26/en-US/AnimatingObjects/Sequencer/ReferenceEditor/

initial scene state and properties can be set at later times using keyframes to set the subsequent changes to the animation at specific times. In addition, users can control attributes of the timing and the interpolation between keyframes in order to create smooth animations.

The following is from the Unreal Engine documentation that describes how keyframes are added at subsequent points on the timeline.[683]

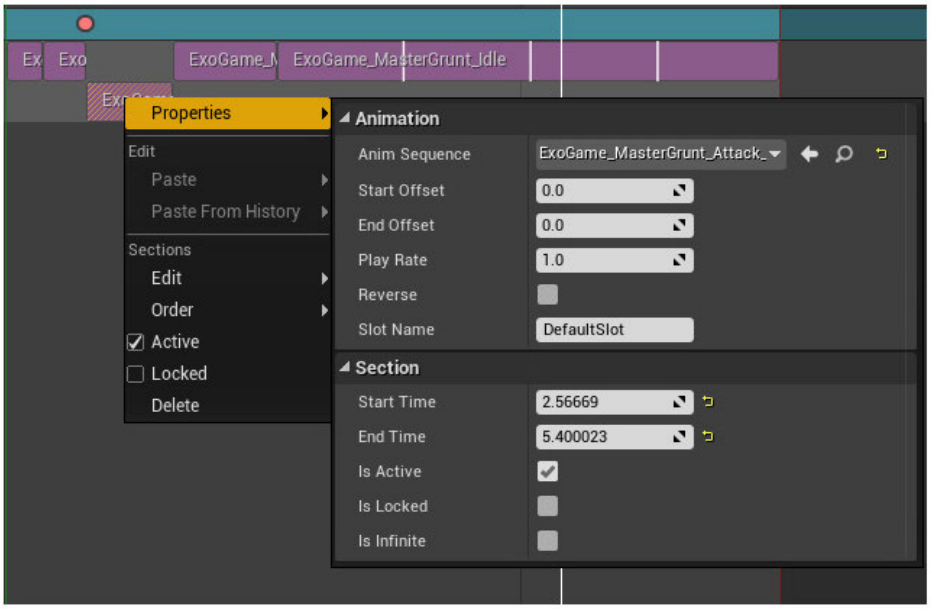




[683] https://docs.unrealengine.com/4.26/en-US/AnimatingObjects/Sequencer/ReferenceEditor/

Above, right-clicking a keyframe for a Transformation track gives the ability to change the key interpolation type as well as other options that can be adjusted from the **Properties** rollout, which updates based on the type of content your Right-click on.

Below, right-clicking on an animation assigned to an Animation Track and selecting properties gives us the ability to change the currently assigned animation as well as adjust properties related to the animation itself, such as start/end offsets, play rate, and more.



By setting keyframes on the Sequencer timeline, the Epic designers and implementers of the Ariana Grande concerts were able to save both the initial scene state as well as subsequent changes at specific times in the animation that cover a time period of the recorded experience.

Below is a representative Unreal Engine asset file as well as blueprints that are examples of the recorded experience file, retrieving the recorded experience file, defining the movements and attributes of objects, as well as playing back the recorded experience file.

[redacted]

[redacted]

[redacted]

[redacted]

[redacted]

[redacted]

[redacted]	[redacted]	[redacted]
[redacted]	[redacted]	[redacted]
[redacted]	[redacted]	
[redacted]	[redacted]	[redacted]
[redacted]	[redacted]	[redacted]
[redacted]	[redacted]	
[redacted]	[redacted]	[redacted]
[redacted]	[redacted]	[redacted]

17.3.1 Epic's Non-Infringement Contentions

Epic in their Third Amended Non-Infringement Chart for '605 (Exhibit K, page 1), states the following:

Epic denies that the Ariana Grande Concert involved "retrieving a recorded experience file from the memory, the recorded experience file having been generated by saving an initial scene state and saving subsequent changes and respective times during a time period of the recorded experience." Every instance of the Ariana Grande Concert, including every repeat performance, was generated using identical programming of an animation sequence involving hundreds of individually animated objects that players in each instance experienced independently from other instances. There was no initial "recorded experience" "during a time period" of which an initial scene state and subsequent changes were saved. Furthermore, the concert music was a sound recording implemented in the animation sequence; no music was recorded during the actual concert event. In addition, there was no "scene state" information saved that included avatars.

I disagree with Epic on their other assertions for the following reasons:

1) The animated concert sequences of the Ariana Grande concerts were recorded in advance of the first concert for later playback in each of the five broadcasts. Mr. Mark Imbriaco described that the triggering event starts a sequence "*refers to the preprogrammed – think of it as the choreography of the event that is predesigned that exists on the server and the client already."*[684] He goes on to say, "*When the live events start, they're not truly live. I mean, they are live, they're – but they're being driven by this preprogrammed sequence of movements that are designed using a technology called the sequencer.*"[685] Mr. Peter Axt described that the playlists for the concert was the same for each of the five performances.[686] The Epic Games technology of the Unreal Engine Sequencer, combined with the Fortnite environment provided, "A method of playing back a recorded experience in a virtual worlds system."
2) The concert sequences are stored in a recorded experience file for later playback (e.g., [redacted] (bates EPIC-SRC3).
3) As described in section Claim 1[iii] for the '605 patent, the attributes of objects in the scene are stored using the Unreal Engine Sequencer as the initial scene state. This includes data that describes the positions of objects like the jeweled hammer and the

[684] Deposition Testimony of Mr. Mark Imbriaco, 314:11-25
[685] Deposition Testimony of Mr. Mark Imbriaco, 315:1-5
[686] Deposition Testimony of Mr. Peter Axt, 298:21-299:3

position and poses of the mannequin of Ariana Grande at the time of the start of the scene (e.g., the initial scene state that happens at the beginning of the first scene.)

4) Subsequent keyframes store attributes of objects in the scene at subsequent times and depict the values of attributes at a given specific time period. This includes data that describes the positions of objects like the jeweled and the positions and poses of the Ariana Grande mannequin. The Sequencer software will then interpolate the change in the attributes from the initial scene state to the first keyframe as well as between any two given keyframes in order to generating a smooth and continuous animation between sets of keyframes.[687]
5) When the recorded experience is played back during run-time, the Fortnite software that is running on the client device will render the initial scene state as well as all subsequent changes over time.[688]
6) The binary asset file contains the data that determines the movements and motions of objects in the scene such as the jeweled hammer and the Ariana Grande avatar is called [redacted] (bates EPIC-SRC3) and was played back in the same way for each of the five Ariana Grande concert performances that took place between August 8th, 2021 and August 25th, 2021.[689] [690]
7) The music of Ariana Grande was pre-recorded and was played back the same way for each of the five performances.[691] In this way, the music was also part of a recorded experience file, that resided in memory, and was reproduced in the same way for each of the five Ariana Grande performances.[692]

---

[687] https://docs.unrealengine.com/5.1/en-US/creating-animation-keyframes-in-unreal-engine/
[688] Deposition Testimony of Mark Imbriaco, 118:8-119:23
[689] https://www.epicgames.com/site/en-US/news/ariana-grande-steps-into-the-metaverse-as-the-headliner-for-fortnites-rift-tour
[690] EPIC00014958 – Buffet - Run of Show
[691] Deposition Testimony of Mr. Peter Axt, 298:21-299:3
[692] *Id.*

17.4 Claim 1[iv]

"playing back the recorded experience file by rendering, for display by the at least one client device, objects of the initial scene state in the new instance, including one or more avatars, and rendering updates to the initial scene state based on the subsequent changes over the time period; and"

"playing back the recorded experience file by rendering, for display by the at least one client device,"

As described previously, the Unreal Engine Sequencer allows for the creation of animations (e.g., creating a recorded experience and then saving a recorded experience in a file (e.g., [redacted], bates EPIC-SRC3.)) The Fortnite environment that is running on the client device is responsible playing back and rendering the recorded experience file by getting the assets from the server, loading it from disk, to memory, to the processor and GPU on the client device for rendering.[693] As discussed previously, the client device can be a PC, a Mac, or one of several gaming consoles including Sony PlayStation, Microsoft Xbox, Nintendo Switch, as well as Android devices and iOS devices.[694] [695]

"objects of the initial scene state in the new instance, including one or more avatars,"

Objects of the initial scene state simply mean objects that are present initially in the recorded experience file. An example of this could be the big jeweled hammer that is displayed at the beginning of the Escher scene.[696] This object is in the common space because there are three-dimensional blocking volumes that surround the hammer so that the attendees' avatars are

[693] Deposition Testimony of Mark Imbriaco, 118:8-119:23
[694] https://www.epicgames.com/help/en-US/fortnite-c5719335176219/technical-support-c5719372265755/what-platforms-or-devices-are-compatible-with-fortnite-a5720385202331
[695] https://www.fortnite.com/mobile
[696] https://www.youtube.com/watch?v=Ae75UuXQ0eU (time 13:21)

unable to get to it.[697] This object is in the recorded experience file because the hammer moves in the same way each time the Ariana Grande concert is broadcast.[698]

Also, as mentioned previously, the Court has determined that the initial scene state does not have to include avatars, but may include avatars. In other words, the Court has determined that the initial scene state may include avatars but it is not required that it include avatars.[699]

and rendering updates to the initial scene state based on the subsequent changes over the time period;

As described above in section Claim 1[iii], the attributes of objects in the scene are stored using the Unreal Engine Sequencer as the initial scene state.[700] Subsequent keyframes store attributes of objects in the scene at subsequent times and depict the values of attributes at a given

[697] https://www.youtube.com/watch?v=Ae75UuXQ0eU (time 13:24 – 13:28)
[698] Deposition Testimony of Peter Axt, 298:21-299:3
[699] Courts Order Regarding Claims Constructions, page 13.
[700] Deposition Testimony of Peter Axt, 164:1-18, 217:6-220:25

specific time period.[701] The Sequencer software will then interpolate the change in the attributes from the initial scene state to the first keyframe as well as between any two given keyframes in order to generating a smooth and continuous animation.[702] When the recorded experience is played back at run-time, the Fortnite software running on the client device will render the initial scene state as well as all subsequent changes over time.[703]

The animated concert sequences of the Ariana Grande concerts were recorded in advance for later playback in a virtual world.[704] [705] It should be noted that the Court has already determined that the initial scene state does not require the presence of avatars. Avatars may simply be included, but they are not required.[706] The recorded experience included objects (which could but was not required to include avatars) rendered over a time period.[707]

Below is a representative Unreal Engine asset file and blueprints that are responsible for the recorded experience file, retrieving the recorded experience file, defining the movements and attributes of objects, and playing back the recorded experience file.

[redacted]

[redacted]

[redacted]

[701] *Id.*
[702] https://docs.unrealengine.com/5.1/en-US/creating-animation-keyframes-in-unreal-engine/
[703] Deposition Testimony of Mr. Mark Imbriaco, 118:8-120:16
[704] Deposition Testimony of Mr. Mark Imbriaco, pages 314:11–315:14
[705] EPIC-14957 (Run of Show for the Ariana Grande Scott concert
[706] Court Order Regarding Claims Construction, page 13
[707] EPIC-14957 (Run of Show for the Ariana Grande concert

[redacted]

[redacted]

[redacted]

[redacted]	[redacted]	[redacted]
[redacted]	[redacted]	[redacted]
[redacted]	[redacted]	[redacted]
[redacted]	[redacted]	[redacted]
[redacted]	[redacted]	[redacted]
[redacted]	[redacted]	[redacted]
[redacted]	[redacted]	[redacted]
[redacted]	[redacted]	[redacted]

17.4.1 Epic's Non-Infringement Contentions

Epic in their Third Amended Non-Infringement Chart for '605 (Exhibit K, page 2), states the following:

Epic denies that the Ariana Grande Concert involved "playing back the recorded experience file by rendering, for display by the at least one client device, objects of the initial scene state in the new instance, including one or more avatars, and rendering updates to the

initial scene state based on the subsequent changes over the time period." Every instance of the Ariana Grande Concert, including every repeat performance, was generated using identical programming of an animation sequence involving hundreds of individually animated objects that players in each instance experienced independently from other instances. At no point was a "recorded experience file" with "scene state" information saved during a "time period" of an earlier "recorded experience" used to play back a saved version of a previously instantiated concert performance. In addition, there was no "scene state" information saved that included avatars. Further, no instance of the Ariana Grande Concert contained "one or more avatars" that had been present in any prior instance of the concert; Ariana Grande herself was a preprogrammed nonplayer-character ("NPC"), not an avatar.

I disagree with Epic on their other assertions for the following reasons:

1) The animated concert sequences of the Ariana Grande concerts were recorded in advance for later playback in a virtual world.[708] [709]
2) The concert sequences are stored in a recorded experience file for later playback (e.g., [redacted](bates EPIC-SRC3).
3) As described in section Claim 1[iii] for the '605 patent, the attributes of objects in the scene are stored using the Unreal Engine Sequencer as the initial scene state. This includes data that describes the positions and poses of the Ariana Grande mannequin at the initial time (e.g., the beginning of the first scene.)[710]
4) Subsequent keyframes store attributes of objects in the scene at subsequent times and depict the values of attributes at a given specific time period. This includes data that describes the positions and poses of the Ariana Grande mannequin at subsequent times. The Sequencer software will then interpolate the change in the attributes from the initial scene state to the first keyframe as well as between any two given keyframes in order to generating a smooth and continuous animation.[711]

[708] Deposition Testimony of Mr. Mark Imbriaco, 314:11–315:14
[709] EPIC-14957 Run of Show for the Ariana Grande concert
[710] Deposition Testimony of Mr. Peter Axt, pages 164:1-18, 217:6-218:25
[711] https://docs.unrealengine.com/5.1/en-US/creating-animation-keyframes-in-unreal-engine/

5) When the recorded experience is played back at run-time, the Fortnite software running on the client device will render the initial scene state as well as all subsequent changes over time.[712]
6) This binary file has the data that determines the movements and motions of the Ariana Grande avatar that was played back in the same way for each of the five Ariana Grande concert performances that took place between August 6th, 2021 and August 8th, 2021.[713] [714] [715]
7) The music of Ariana Grande was pre-recorded and was played back the same way for each of the five performances.[716] In this way, the music was part of the recorded experience file, that resided in memory, and was reproduced in the same way for each of the five Ariana Grande performances.[717]
8) Epic states, but the claim limitation does not require that, "*At no point was a "recorded experience file" with "scene state" information saved during a "time period" of an earlier "recorded experience" used to play back a saved version of a previously instantiated concert performance.*

17.5 Claim 1[v]

"automatically transporting the one or more avatars to a different new instance of the scene, upon occurrence of threshold event, wherein the threshold event comprises when a maximum capacity of avatars has been reached in the new instance of the scene."

The matchmaking process as described above is responsible for placing players (or attendees in the case of the Ariana Grande concerts) into one instance of the virtual world.[718]

---

[712] Deposition Testimony of Mr. Mark Imbriaco, 118:8-120:16
[713] https://www.fortnite.com/news/astronomical
[714] Deposition Testimony of Mr. Mark Imbriaco, 314:11–315:14
[715] EPIC-14957 Run of Show for the Ariana Grande concert
[716] Deposition Testimony of Mr. Peter Axt, 298:21-299:3
[717] *Id.*
[718] Deposition Testimony of Mr. Peter Axt, 78:12-15

The matchmaking process ensures that the number of players in an instance of the scene is not greater than the maximum number of players allowed in that new instance.[719] The matchmaking process keeps track of how many players have been assigned to a given new instance and when the maximum number of players that are assigned to a given instance equals the maximum number of players that are allowed for any given instance (e.g., the [redacted]).[720] When the number of players assigned to a given new instance equals the maximum number of players allowed for a given instance, that is threshold event that results in new additional players being put into a different new instance of the scene.

On the client side, the various technologies of the Master Control Program (MCP), and Matchmaking Service (MMS) ensure that the client device gets connected to a specific dedicated server that has capacity and only needs to handle the state of the virtual world for the much smaller subset of attendees that are associated with that dedicated server.[721]

The first step is that when the user is in the Lobby, they can utilize the user interface to [redacted] request to matchmake [redacted]

[719] Deposition Testimony of Mr. Peter Axt, 249:3-250:19
[720] Deposition Testimony of Mr. Peter Axt, 249:3-250:19
[721] Deposition Testimony of Peter Axt at 34
[722] Deposition Transcript of Peter Axt at 236:6-21
[723] *Id.*
[724] *Id.*
[725] Deposition Transcript of Peter Axt at 237:14-19
[726] Deposition Transcript of Peter Axt at 237:11-14
[727] Deposition Transcript of Peter Axt at 239:2-12

[redacted]

Given that there was a maximum number of [redacted] per dedicated server session for the Travis Scott and Ariana Grande concert, players could only be assigned to a dedicated server session if there were fewer than [redacted].[730] It is not possible or a dedicated server to receive more than its configured maximum number of players.[731] The matchmaking service knows how many available slots are in the dedicated server session.[732] As the matchmaking service is responsible for assigning tickets to a game session, it is aware of how many tickets it has assigned to each dedicated server so that it will not over-populate any given dedicated server.[733] Once a given dedicated server is full (e.g., [redacted]), the matchmaking is complete for that dedicated server.[734]

[redacted]

[728] Deposition Transcript of Peter Axt at 241:17-242:11
[729] Deposition Transcript of Peter Axt at 243:9-244:21
[730] Deposition Transcript of Peter Axt at 245:23-246:22
[731] Deposition Transcript of Peter Axt at 247:5-21
[732] Deposition Transcript of Peter Axt at 249:17-250:3
[733] Deposition Transcript of Peter Axt at 250:4-250:19
[734] Deposition Transcript of Peter Axt at 251:11-252:22
[735] Deposition Transcript of Peter Axt at 252:3-22
[736] *Id.*
[737] Deposition Transcript of Peter Axt at 255:2-18

[redacted]

Mr. Mark Imbriaco explains that information about the world is stored using a replication graph. He describes the replication graph as being *"the piece of technology that is used to determine what changes in game state are relevant to each player at a point in time, and provide those updates to players, as needed, so that they have a reasonably accurate representation of the game state that's relevant to them"* and that the replication graph *"includes position of objects, characters, and other parts of game state."* [739] The game state information is delivered through the replication graph and this would include the four accused events.[740] Mr. Imbriaco say that it is the server that distributes game-state changes to the clients.[741] [742] The Fortnite environment that is running on the client device is responsible playing back and rendering by getting any needed the assets from server (via the replication graph), and rendering it on the client device.[743] As discussed previously, the client device can be a PC, a Mac, or one of several gaming consoles including Sony PlayStation, Microsoft Xbox, Nintendo Switch, as well as Android devices and iOS devices.[744] [745]

In the way described above, Epic Games is able to play back information that is captured in the recorded experience file so that the client devices can render objects of the initial scene state that were present in the instances. The replication graph allows for sending the required information so that the clients can render updates to the initial scene state given the subsequent changes that are present in the replication graph over any time period.

---

[738] Deposition Transcript of Peter Axt at 256:5-24
[739] Deposition Testimony of Mark Imbriaco, 113:7-13, 113:17-18
[740] Deposition Testimony of Mark Imbriaco, 122:7-13
[741] Deposition Testimony of Mark Imbriaco, 126:11-15
[742] Deposition Testimony of Mark Imbriaco, 142:2-11
[743] Deposition Testimony of Mark Imbriaco, page 118
[744] https://www.epicgames.com/help/en-US/fortnite-c5719335176219/technical-support-c5719372265755/what-platforms-or-devices-are-compatible-with-fortnite-a5720385202331
[745] https://www.fortnite.com/mobile

It is the Epic technologies of Autoscaling, Dedicated Servers, Master Control Program and the Matchmaking Service that are described above that are responsible for making sure that fleets of new instances are available to host the millions of users that attended the Ariana Grande concerts and that participants get put into new instances, but no more than the maximum number of participants allowed. This article states that the concert was broadcast five times over three days and there were [redacted] players that participated in the concerts.[746] Given that there were [redacted] that were in each instance[747], this equates to approximately [redacted] instances that were needed to support the Ariana Grande concert.

Below are representative source code files and blueprints that are responsible for limiting the number of avatars in any given instance and for automatically transporting avatars from the lobby to one of the new instances of the scene when the number of avatars in the instance that is currently being filled reaches a maximum capacity.

[redacted]

[redacted]

[redacted]

[redacted]

[746] https://www.metaversemarcom.io/post/top-10-most-popular-metaverse-concerts
[747] Deposition Testimony of Peter Axt, 246:15-17.

[redacted]

[748] Deposition Testimony of Peter Axt, 246:15-17
[749] Deposition Testimony of Peter Axt, 311:18-312:6

[redacted]

[750] https://docs.unrealengine.com/4.26/en-US/API/Plugins/OnlineSubsystemUtils/APartyBeaconHost/

Name	Bates	Description
[redacted]	[redacted]	[redacted]

17.5.1 Epic's Non-Infringement Contentions

Epic in their Third Amended Non-Infringement Chart for '605 (Exhibit K, pages 2 - 3), states the following:

Epic denies that the Ariana Grande Concert involved "automatically transporting the one or more avatars to a different new instance of the scene, upon occurrence of a threshold event, wherein the threshold event comprises when a maximum capacity of avatars has been reached in the new instance of the scene."

Epic denies that the Ariana Grande Concert involved "automatically transporting the one or more avatars to a different new instance of the scene." No instance of the Ariana Grande Concert contained "one or more avatars" that had been present in any prior instance of the concert and rendered from a recorded experience file, nor were such avatars transported to a new instance of a scene. Ariana Grande herself was a preprogrammed NPC, not an avatar. Identical copies of the Ariana Grande NPC were created in every instance; the Ariana Grande NPC was not "transported" between instances. Further, player avatars could not travel between instances of the Ariana Grande Concert, when a maximum capacity has been reached or otherwise; each player was assigned to an instance by the Matchmaking Service ("MMS") before the instance and the player's avatar were generated and player avatars remained in that instance for the entire performance.

Epic further denies that avatars were transported "upon occurrence of a threshold event, wherein the threshold event comprises when a maximum capacity of avatars has been reached in the new instance of the scene." The instances of the game environment hosting the Ariana Grande Concert were generated using MMS, a service that detected and matched a pool of available players according to different attributes of the players, the client devices, and their connections, before the game environment, and the player's avatars, were generated; a player avatar could not enter an instance of the Ariana Grande Concert without first being matched with a pool of other players. Moreover, the MMS would match and populate instances at numbers below the maximum threshold of avatars for each instance.

I disagree with Epic on their other assertions for the following reasons:

1) The matchmaking process as described above is responsible for placing players (or attendees in the case of the Ariana Grande concerts) into one instance of the virtual world.[751] The matchmaking process ensures that the number of players in an instance of the scene is not greater than the maximum number of players allowed in that new instance.[752]
2) The matchmaking process keeps track of how many players have been assigned to a given new instance and when the maximum number of players that are assigned to a given instance equals the maximum number of players that are allowed for any given instance (e.g., the [redacted]).[753] When the number of players assigned to a given new instance equals the maximum number of players allowed for a given instance,

[751] Deposition Testimony of Mr. Peter Axt, 78:12-15
[752] Deposition Testimony of Mr. Peter Axt, 249:3-250:12
[753] *Id.*

that is threshold event that results in new additional players being put into a different new instance of the scene.

3) Avatars were automatically transported from the Lobby to the first scene. The attendees had the same avatar while in the Lobby as they did when they were in the first scene. The attendees were automatically transported out of the Lobby and into a new instance upon the occurrence of a threshold event. The threshold event is when a previous instance is full. In the case of Ariana Grande, there was a maximum of [redacted] that were supported by each instance.[754] The attendees were automatically transported out of the Lobby and into the first scene.
4) Epic states that "player avatars could not travel between instances of the Ariana Grande Concert." The claim limitation does not require that avatars be able to travel between instances of the Ariana Grande Concert.

17.6 Claim 2

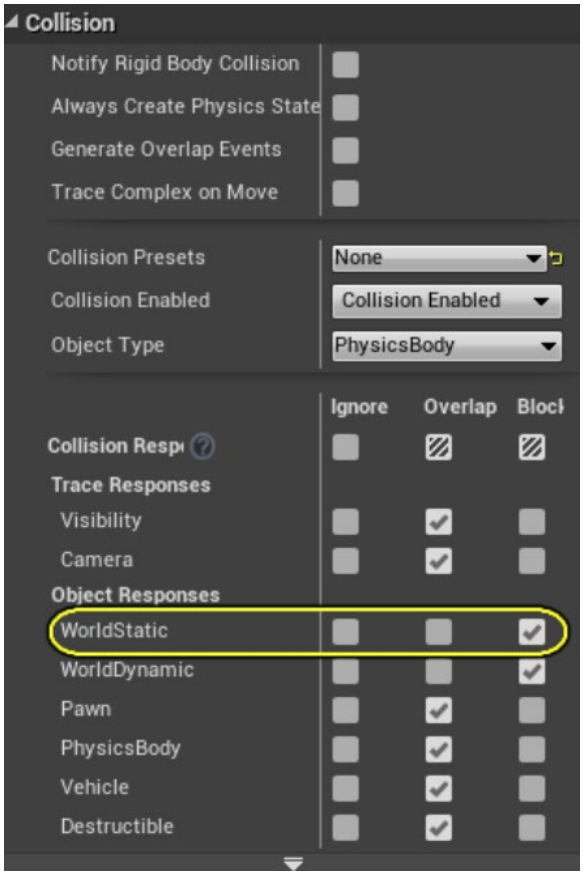
"The method of claim 1, wherein movement within the new instance by the one or more avatars associated with at least one client device is limited by objects of the recorded experience."

The way that Fortnite is designed (as well as the design of the Ariana Grande concerts), each individual player is associated with only one client device that they use to interface with Fortnite and the concert. In addition, each individual player is associated with only one avatar. Also, as discussed previously in this report, Epic uses three-dimensional blocking volumes in order to limit movement of avatars.

[754] Deposition Testimony of Peter Axt, page 246:15-17

The Unreal Engine documentation titled Collision Overview describes in detail how blocking works within Unreal Engine.[755] The document states that "Blocking will naturally occur between two (or more) Actors set to Block."

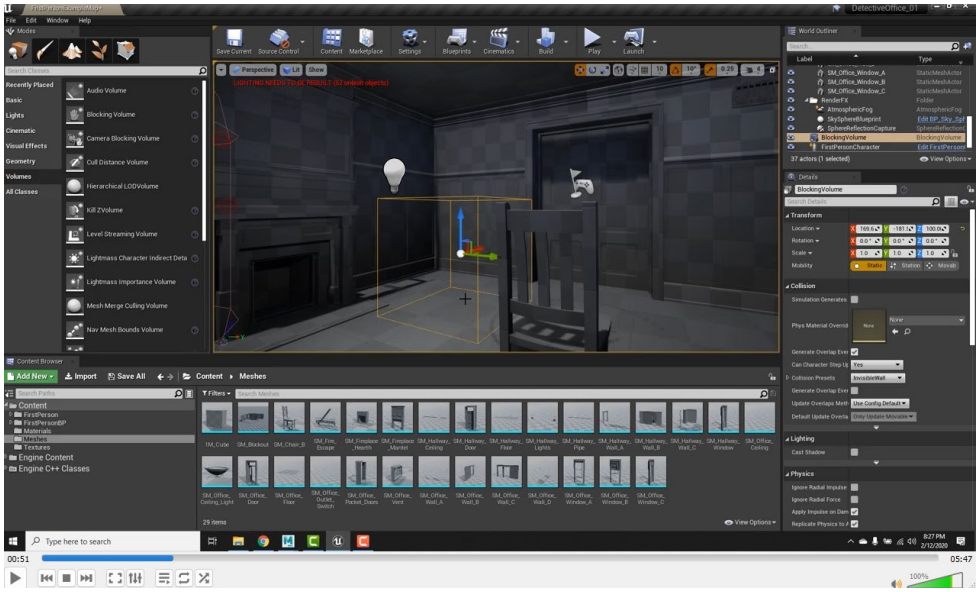
This Unreal Engine dialog box below allows the user of Unreal Engine to set the attributes of a given object to Block, that is what is desired. Additionally, Unreal Engine supports the automatic generation of hit events when a collision takes place with events such as ReceiveHit or OnComponentHit. In this way, the designers and developers of the Ariana Grande concerts created blockers (that were included into the sequence and were therefore part of the recorded experience file) that limited the movement of avatars within the instance.



[755] EPIC-20389

In the same way that was described above for the infringement analysis of the Ariana Grande Scott concerts on '071 patent, participant avatars could move about within the concert venue, but only within defined limits of that venue.[756] Blocking volumes defined an area or stage (e.g., the common area) that represented an area that the participant avatars could not move to. The blocking volumes are invisible three-dimensional objects that set limits on where participant avatars can travel.[757]

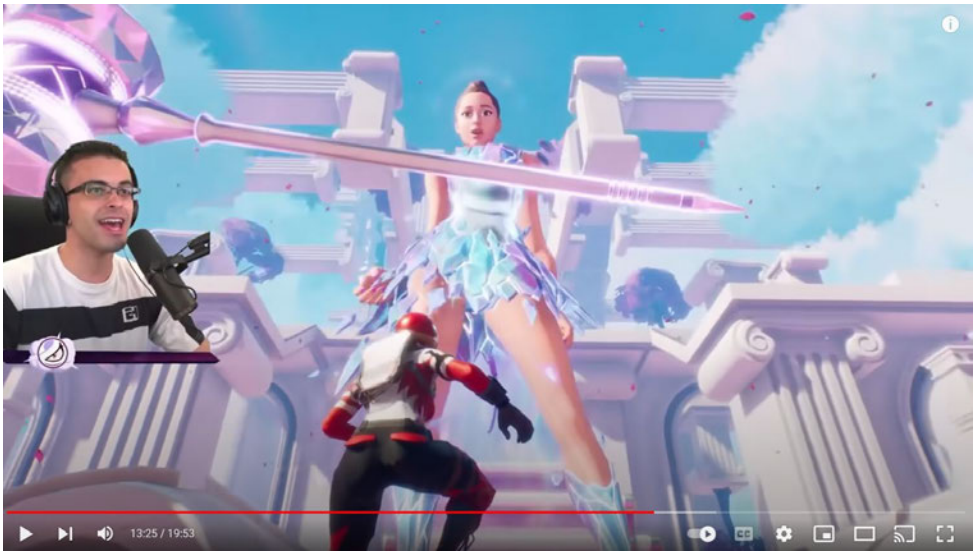
This video that is in this footnote shows how users can setup blocking volumes in Unreal Engine.[758] If one looks at time 0:25 – 1:30, the instructor shows how to utilize three-dimensional blocking volumes can be setup to limit the motion of players within the scene.



Here is a video that shows how the blocking volumes prevented user avatars from entering the common space. https://www.youtube.com/watch?v=Ae75UuXQ0eU If one looks

[756] Deposition Testimony of Peter Axt, 158:4-160:22
[757] EPIC-00020291
[758] EPIC-00020405.mp4

at times 13:21 to 13:36, one will see how the blocking volumes prevented user avatars from entering the area right around the mannequin of Ariana Grande (e.g., the common space.)



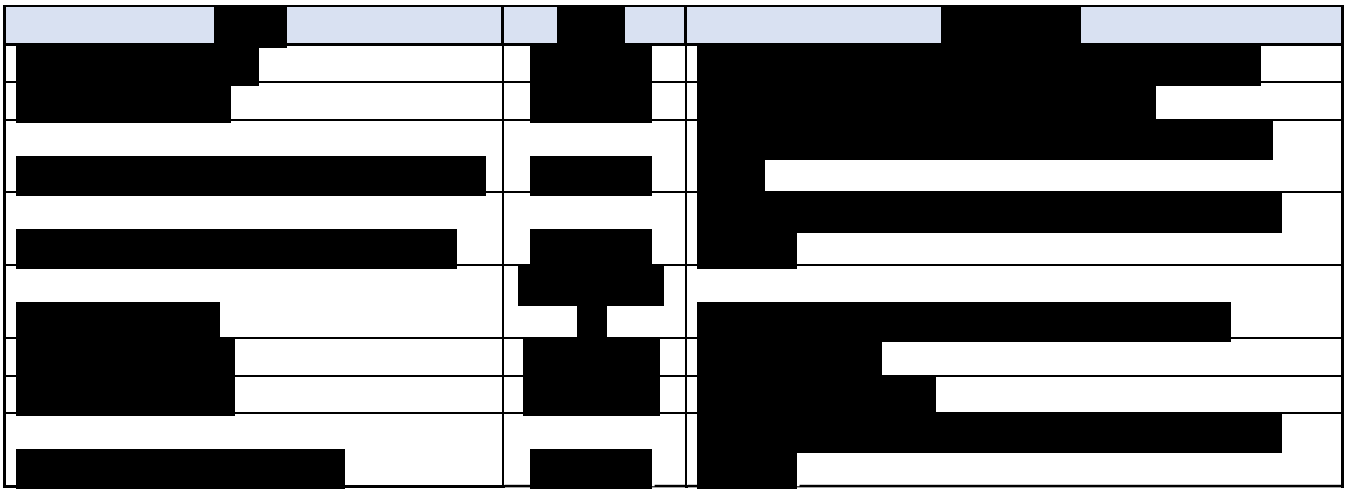
Below are representative source code files and blueprints that are responsible for both moving the avatars that are associated with players (and client devices), limiting the movement of avatars that are controlled by players, and the blueprint for the recorded experience file that contains blocking volumes.

[redacted]

[redacted]

[redacted]

[redacted]



17.6.1 Epic's Non-Infringement Contentions

Epic in their Third Amended Non-Infringement Chart for '605 (Exhibit K, page 3), states the following:

Epic denies that the Ariana Grande Concert included that "movement within the new instance by the one or more avatars . . . is limited by objects of the recorded experience." No

[759] https://www.youtube.com/watch?v=Ae75UuXQ0eU

objects of a recorded experience were present in any instance of the Ariana Grande Concert. No instance of the Ariana Grande Concert contained "one or more avatars" that had been present in any prior instance of the concert and rendered from a recorded experience file. The Ariana Grande NPC's movement was not limited by any object. Further, player avatars' movements were limited by objects and map settings in a newly generated instance of the Ariana Grande Concert, not by objects from a recorded experience.

I disagree with Epic on their other assertions for the following reasons:

1) As discussed previously in this report, Epic uses three-dimensional blocking volumes in order to limit movement of avatars. The Unreal Engine documentation titled Collision Overview describes in detail how blocking works within Unreal Engine.[760] The document states that "Blocking will naturally occur between two (or more) Actors set to Block." In this way, the designers and developers of the Ariana Grande concerts created blockers (that were included into the sequence and were therefore part of the recorded experience file) that limited the movement of avatars within the instance.[761]
2) The three-dimensional blocking volumes were part of the recorded experience that is captured in the file [redacted] (bates EPIC-SRC3). [redacted] When looking at this object in the Unreal Engine Editor, we see that this same object is of type Blocking Volume when looking at the details pane. Lastly, in the video at time, 13:30 – 13:35 we see player avatars unable to get past the blocker that surrounds this pillar because of the blocking volume just discussed.[762]

[760] EPIC-20389
[761] Deposition Testimony of Peter Axt, 158:2-160:22
[762] https://www.youtube.com/watch?v=Ae75UuXQ0eU

17.7 Claim 5

"The method of claim 1, wherein the recorded experience file is not modifiable by events occurring during playback of the recorded experience."

As discussed previously, the Ariana Grande event took place five times from Aug 6th, 2021 through Aug 8th, 2021. The recorded experience file was the same in all Ariana Grande events.[763] While the users (participant avatars) could perform new actions in the virtual world, but those new actions would not modify the recorded file itself.

In reviewing the Epic source code in February and March of 2023, I found no evidence that the recorded experience file was modifiable by events occurring during playback of the recoded experience.

In addition, the music of Ariana Grande, which is part of the recorded experience was also not modifiable by events during the playback of the concerts. The music was pre-recorded and the visuals were created to coincide with the music. The music and the visuals were presented the same way each of the five times that the Ariana Grande event was presented.[764]

Below is the relevant recorded experience file, source code files, and blueprints that define the recorded experience file and control the playback of the recorded experience during run-time of the Ariana Grande concerts.

[redacted]

[763] Deposition Testimony of Mr. Peter Axt, 298:19-25
[764] Deposition Testimony of Mr. Peter Axt, 298:19-25

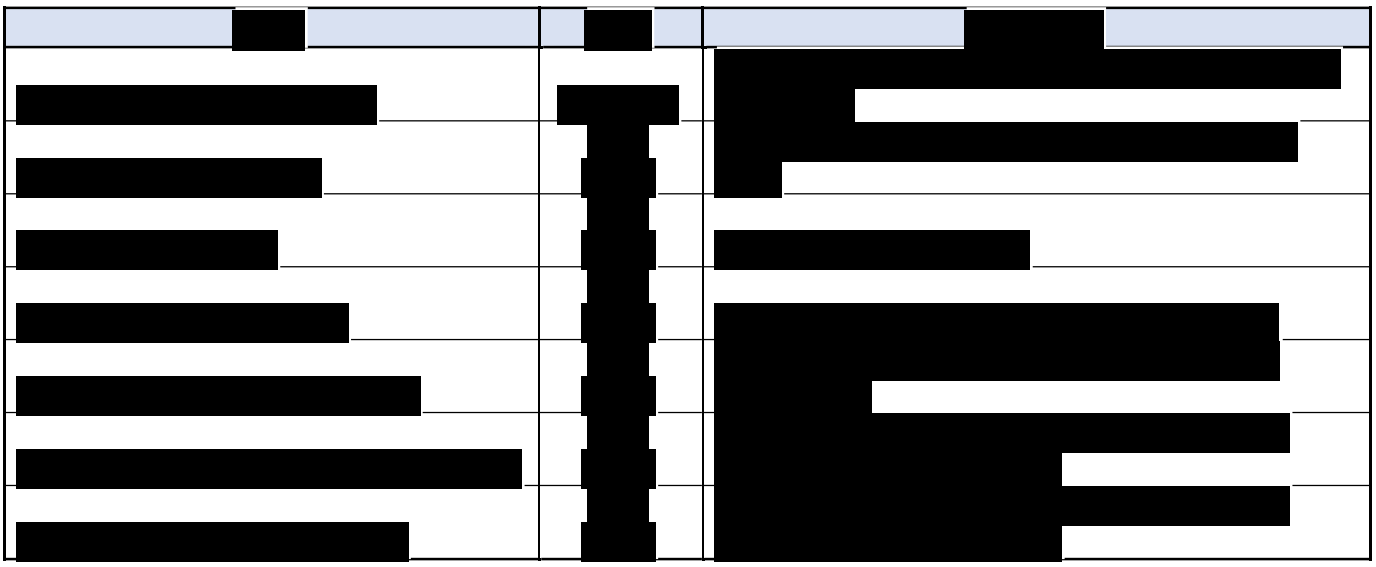
[redacted]

[redacted]

[redacted]

[redacted]

[redacted]



17.7.1 Epic's Non-Infringement Contentions

Epic in their Third Amended Non-Infringement Chart for '605 (Exhibit K, pages 3 - 4), states the following:

Epic denies that the Ariana Grande Concert included that "the recorded experience file is not modifiable by events occurring during playback of the recorded experience." The Ariana Grande Concert did not involve playing back a recorded experience file; rather, it was an interactive, pre-programmed sequence animated in real-time. Accordingly, there was no recorded experience file that was capable or incapable of being modified.

I disagree with Epic on their other assertions for the following reasons:

1) The Ariana Grande concert did playback a recorded experience file. The recorded experience file was [redacted] (bates EPIC-SRC3.) This file was created using the Unreal Engine Sequencer using keyframe animation.
2) As described in this report, the movements of various elements in the Ariana Grande concerts were preprogrammed using the keyframe animation in the Sequencer of Unreal Engine.[765] These recordings were played back in the same way for each of the five Ariana Grande concerts that took place between August 6th, 2021 and August 8th, 2021.[766] The Unreal engine Sequencer created a recorded experience file [redacted] (bates EPIC-SRC3) that was used to playback the animation of the mannequin of Ariana Grande in the same way each time the concert was broadcast.[767]
3) In reviewing the Epic source code in February and March of 2023, I found no evidence that the recorded experience file was modifiable by events occurring during playback of the recoded experience.

[765] https://docs.unrealengine.com/5.1/en-US/unreal-engine-sequencer-movie-tool-overview/
[766] https://www.epicgames.com/site/en-US/news/ariana-grande-steps-into-the-metaverse-as-the-headliner-for-fortnites-rift-tour
[767] [767] Deposition Testimony of Mr. Peter Axt, 298:19-25

17.8 Claim 8

"The method of claim 1, wherein the new instance of the scene is three-dimensional."

Epic admits that the instances of the Ariana Grande Concert were three-dimensional.[768]

As one can see from the videos of the Ariana Grande concerts, the environment was three-dimensional.[769] Another video of the same event can be found here.[770]

17.8.1 Epic's Non-Infringement Contentions

Epic in their Third Amended Non-Infringement Chart for '605 (Exhibit K, page 4), states the following:

Epic admits that the instances of the Ariana Grande Concert were three-dimensional.

I agree with Epic that, "*that the instances of the Ariana Grande Concert were three-dimensional"* and the videos from the events prove that as well.

[768] Third Amended Non-Infringement Chart of '605 Patent – Exhibit K – Page 4
[769] https://www.youtube.com/watch?v=Ae75UuXQ0eU
[770] https://www.youtube.com/watch?v=RiM0moNk74o

Screenshot from video showing the Ariana Grande concert was three-dimensional.

https://www.youtube.com/watch?v=Ae75UuXQ0eU

Screenshot from video showing the Ariana Grande concert was three-dimensional.

https://www.youtube.com/watch?v=RiM0moNk74o

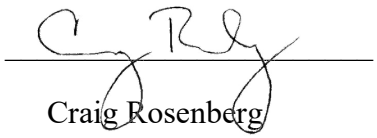
18 CONCLUSION

For the reasons provided above, it is my opinion that claims 8 and 10 of the ‘071 patent and claims 2, 5, and 8 of the ‘605 patent are infringed on by Epic Games, Inc., by the creation, production, and presentation of the four accused events (Marshmello, Star Wars: The Rise of Skywalker, Travis Scott, and Ariana Grande) and the various Epic technologies that were used in the creation, production, and the presentation of the four accused events. These technologies include any custom software development associated with the creation, production, and presentation of the four accused events, as well as the Fortnite gaming environment and Unreal Engine that was also used in the creation, production, and presentation of the four accused events.

I declare all statements made herein of my knowledge are true, and that all statements made on information and belief are believed to be true, and that those statements were made with the knowledge that willful false statements and the like so made are punishable by fine, imprisonment, or both, under Section 1001 of Title 18 of the United States Code.

Respectfully submitted,

Date: April 14th, 2023

By: 

Craig Rosenberg

Facts & Data Considered – Appendix B

Legal Documents

1. 2021.06.11[ECF 1] Complaint for Patent Infringement with exhibits
2. 2022.04.28 [ECF 72] Defendant Epic Games, Inc's Opening Claim Construction Brief
3. 2022.04.28 [ECF 76] Utherverse Gaming LLC's Opening Claim Construction Brief
4. 2022.10.20 [ECF 133] Order Re: Claims Construction
5. 2022.11.15 [ECF 146] Order Denying Motion for Reconsideration
6. Claim Construction Hearing Transcript
7. 2021.11.02 Plaintiff's Disclosure of Asserted Claims and Infringement Contentions Under LPR 120
8. 2023.03.03 Plaintiff's First Amended Disclosure of Asserted Claims and Infringement Contentions Under LPR 120
9. 2023.03.24 Plaintiff's Second Amended Disclosure of Asserted Claims and Infringement Contentions Under LPR 120
10. 2021.12.01 Defendant Epic Games, Inc's Disclosure of Non-Infringement Contentions Pursuant to L.P.R. 121
11. 2022.05.09 Defendant Epic Games, Inc.'s Frist Amended Non-Infringement Contentions
12. 2022.06.15 Defendant Epic Games, Inc.'s Second Amended Non-Infringement Contentions
13. 2022.09.02 Defendant's Third Amended Non-Infringement Contentions

Deposition Transcripts

1. 2023.04.23 Deposition Transcript of Peter Axt
2. 2023.04.07 Deposition Transcript of Mark Imbriaco

Bates-Stamped Documents

1. EPIC00000620-621
2. EPIC00000622-706
3. EPIC00000882-886
4. EPIC00000887-891
5. EPIC00000892-897
6. EPIC00000899-907
7. EPIC00000930-949
8. EPIC00014930- [redacted] – Global Run of show
9. EPIC00014931-14933
10. EPIC00014934-14935
11. EPIC00014938-14956
12. EPIC00014957 – [redacted] – Run of Show
13. EPIC00014958 – [redacted] – Run of show
14. EPIC00014959
15. EPIC00017835
16. EPIC00020243
17. EPIC00020244

18. EPIC00020248
19. EPIC00020249
20. EPIC00020264
21. EPIC00020266
22. EPIC00020267
23. EPIC00020272
24. EPIC00020273
25. EPIC00020275
26. EPIC00020276
27. EPIC00020279
28. EPIC00020282
29. EPIC00020285
30. EPIC00020288
31. EPIC00020291
32. EPIC00020380
33. EPIC00020381
34. EPIC00020382
35. EPIC00020383
36. EPIC00020384
37. EPIC00020385
38. EPIC00020387
39. EPIC00020389
40. EPIC00020390
41. EPIC00020394
42. EPIC00020397
43. EPIC00020398
44. EPIC00020400
45. EPIC00020401
46. EPIC00020403
47. EPIC00020405
48. EPIC00020409
49. EPIC00020437-20438
50. EPIC00020439-20459
51. EPIC00020460-20472
52. EPIC00020525-20526
53. EPIC00020542-20544
54. EPIC00020546-20549
55. EPIC00020553-20554
56. EPIC00020555-20574
57. EPIC00021098-21099
58. EPIC00021219-21239
59. EPIC00021255-21257
60. EPIC00021285-21286
61. EPIC00021366-21367
62. EPIC00021368-21369
63. EPIC00021392-21398

64. EPIC00021459
65. EPIC00021478-21479
66. EPIC00021502
67. EPIC00021503-21504
68. EPIC00021510
69. EPIC00021511-21576
70. EPIC00021581-22260
71. EPIC00022261-22451
72. EPIC00022452-22669
73. EPIC00023187-23192
74. EPIC00023193
75. EPIC00023194-23197
76. EPIC00023198-23202
77. EPIC00040437
78. Multi-Instance, Multi-User Animation Platforms, United States Patent 8,276,071 (UG000000001)
79. Method, System and Apparatus of Recording and Playing Back an Experience in a Virtual Worlds System, United States Patent 9,724,605 (UG000000020)
80. https://status.epicgames.com (UG00002581)
81. https://docs.unrealengine.com/4.27/en-US/InteractiveExperiences/Networking/ReplicationGraph/ (UG00002629)
82. https://www.gamesradar.com/fortnite-map/ (UG00002659)
83. https-en.wikipedia.org-wiki-Fortnite_Battle_Royale (UG00002674)
84. https://www.polygon.com/fortnite/2020/4/23/21233294/fortnite-travis-scott-concert-video (UG00002736)
85. https://www.vox.com/culture/2020/4/24/21235196/travis-scott-fortnite-concert-livestream-the-scotts-music-video (UG00002740)
86. https://www.epicgames.com/fortnite/en-US/news/fortnite-presents-the-rift-tour-featuring-ariana-grande (UG00002744)
87. https://www.gamespot.com/articles/fortnite-ariana-grande-rift-tour-watch-it-here/1100-6494920/ (UG00002773)
88. https://fortune.com/2021/10/12/amazon-web-services-aws-offline-error/ (UG00002777)
89. https://www.epicgames.com/fortnite/en-US/news/astronomical (UG00002786)
90. https://www.epicgames.com/help/en-US/fortnite-c75/technical-support-c118/understanding-latency-or-ping-in-fortnite-a7617 (UG00002791)
91. https://www.epicgames.com/fortnite/en-US/rift-tour (UG00002796)
92. https://www.businessinsider.com/screenshots-inside-fortnite-star-wars-rise-of-skywalker-preview-event-2019-12#but-before-we-could-see-the-results (UG00002998)
93. https://www.thegamer.com/fortnite-star-wars-event-live-million-viewers/ (UG00003029)
94. https://www.washingtonpost.com/video-games/2021/08/06/ariana-grande-performs-concert-fortnite/ (UG00003042)
95. https://www.theverge.com/2021/8/1/22602826/ariana-grande-fortnite-rift-tour-concert-epic-games (UG00003057)
96. https://www.theverge.com/2019/2/21/18234980/fortnite-marshmello-concert-viewer-numbers (UG00003075)
97. https://aws.amazon.com/solutions/case-studies/EPICGames/ (UG00003111)

98. https://www.usgamer.net/articles/fortnite-hits-a-concurrent-player-record (UG00003114)
99. https://techcrunch.com/2021/08/09/fortnite-ariana-grande-concert-metaverse/ (UG00003128)
100. https://www.cnet.com/news/fortnite-travis-scott-astronomical-experience-seen-by-almost-28-million-players/ (UG00003148)
101. https://www.hollywoodreporter.com/business/digital/fortnite-presents-rift-tour-ariana-grande-1234994287/ (UG00003161)
102. https://www.tomsguide.com/news/ariana-grande-fortnite-concert-start-time-skins-and-how-to-watch (UG00003206)
103. https://www.pcgamer.com/ariana-grande-fortnite/ (UG00003255)
104. https://twitter.com/FortniteGame/status/1253524351376330752 (UG00003275)
105. https://realsport101.com/fortnite/ (UG00003290)
106. https://www.epicgames.com/site/en-US/news/ariana-grande-steps-into-the-metaverse-as-the-headliner-for-fortnites-rift-tour (UG00003295)
107. https://www.youtube.com/watch?v=Hul56IzA1V8 (UG00003308)
108. https://www.youtube.com/watch?v=HwbScTdNoik (UG00003309)
109. https://www.youtube.com/watch?v=ZyBb6Ha1Un4 (UG00003310)
110. https://www.youtube.com/watch?v=g6WkjnVKAPs (UG00003311)
111. https://www.youtube.com/watch?v=cPYA9Fsmo-0 (UG00003312)
112. https://www.youtube.com/watch?v=gGYElBtjytU (UG00003313)
113. https://www.youtube.com/watch?v=Ae75UuXQ0eU (UG00003314)
114. https://www.youtube.com/watch?v=vJMOfAkrAPo (UG00003315)
115. https://www.youtube.com/watch?v=1UclwReAKR8 (UG00003317)
116. https://www.youtube.com/watch?v=k5EAPwXxcng (UG00003318)
117. https://www.youtube.com/watch?v=NBsCzN-jfvA (UG00003320)
118. https://www.youtube.com/watch?v=wYeFAlVC8qU (UG00003321)
119. (UG00003322)
120. https://www.youtube.com/watch?v=cq3iYRr2q3U (UG00003324)
121. https://www.youtube.com/watch?v=gYP9Katzh0c (UG00003325)
122. https://www.youtube.com/watch?v=p5f-Fc9WVAQ (UG00003326)
123. https://www.youtube.com/watch?v=Z8M72aZsI2E (UG00009488)
124. https://www.youtube.com/watch?v=aNLyly-ft74 (UG00009489)
125. https://www.youtube.com/watch?v=kjj3FwzRQms (UG00009490)
126. https://www.youtube.com/watch?v=Ae75UuXQ0eU (UG00009491)
127. https://www.youtube.com/watch?v=zg6SuCqXI9Q (UG00009492)
128. https://www.youtube.com/watch?v=V9Rnk4sdOl0 (UG00009724)
129. https://www.youtube.com/watch?v=ljJwsr1Qk64 (UG00009735)

Additional Sources

1. Interview with Brian Shuster
2. Examination of Epic Games IDE in Cary, NC (inclusive of Epic source code and other materials specifically identified in the Second Amended Infringement Contentions and/or this Report)
3. Epic Source Code Documentation - EPIC-SRC-0000001-0000170
4. Unreal Engine 4 Documentation https://docs.unrealengine.com/4.27/en-US/
5. Unreal Engine 5.1 Documentation https://docs.unrealengine.com/5.1/en-US/